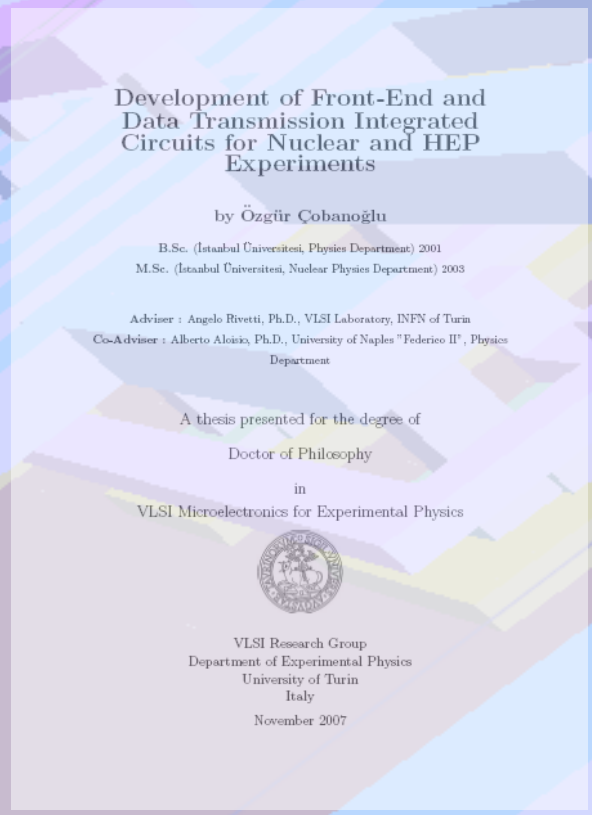
Development of Front-End and Data Transmission Integrated Circuits for Nuclear and HEP Experiments

Özgür Çobanoğlu

Univ. & INFN of Turin/Italy


Development of Front-End and Data Transmission Integrated Circuits for Nuclear and HEP Experiments

by Özgür Çobanoğlu

B.Sc. (İstanbul Üniversitesi, Physics Department) 2001
M.Sc. (İstanbul Üniversitesi, Nuclear Physics Department) 2003

Adviser : Angelo Rivetti, Ph.D., VLSI Laboratory, INFN of Turin
Co-Adviser : Alberto Aloisio, Ph.D., University of Naples "Federico II", Physics Department

A thesis presented for the degree of
Doctor of Philosophy
in
VLSI Microelectronics for Experimental Physics

VLSI Research Group
Department of Experimental Physics
University of Turin
Italy

November 2007

My background studies

- *Graduated from a special profession-oriented high school for **Marin Communication and Ship Electronics** during which I studied discrete electronics for general purpose communication device fixing.*

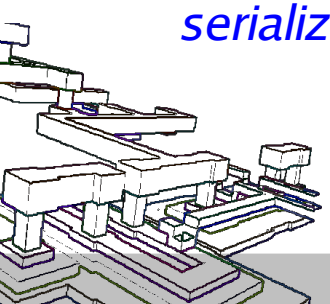
- *Graduated from **Istanbul University (IU), Nuclear Physics Department,** Türkiye. Studied experimental nuclear physics and developed off-line physics analysis software.*

- *Worked on gamma spectrometers AFRODITE in south Africa and GAMMASPHERE at YALE, US, in the name of IU; developed software for every level of off-line spectroscopic analysis during my MSc at IU.*

- *Worked for ALICE at CERN and developed MOOD (Monitor Of On-line Data and Detector Debugger) software for IU.*

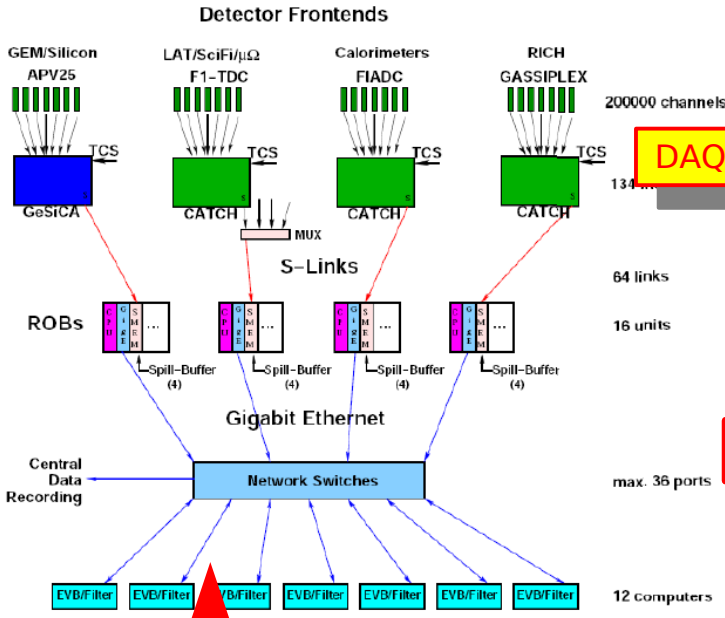
- *Worked for COMPASS (CERN) and contributed to the development of the FE ASIC, the CMAD, for RICH-I detector system within the VLSI group at Turin University.*

- *Worked for S-LHC (CERN) and contributed to the development of CP-PLL based serializer for the GBT13 transceiver within the VLSI group at Turin University.*



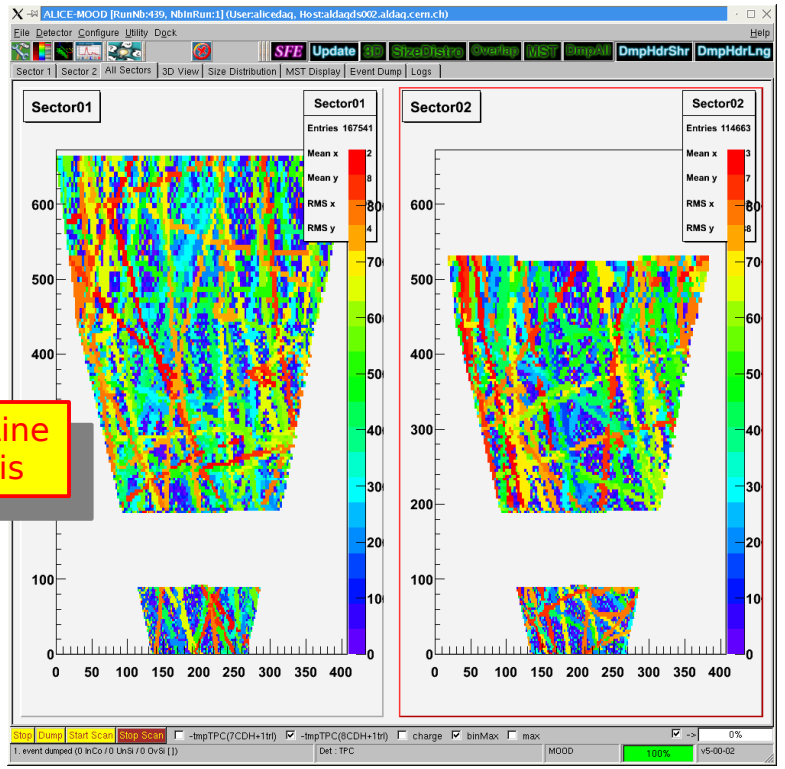
Terminology

- The ***Front-End*** ASIC sees the signal of detection first, performs analog signal processing and pass the to data Read-Out system
- The ***Read-Out*** (RO) System "formats" this data
- The ***Transceiver*** (Tx/Rx within RO) delivers the data to DAQ

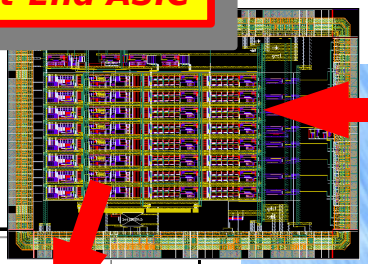


DAQ

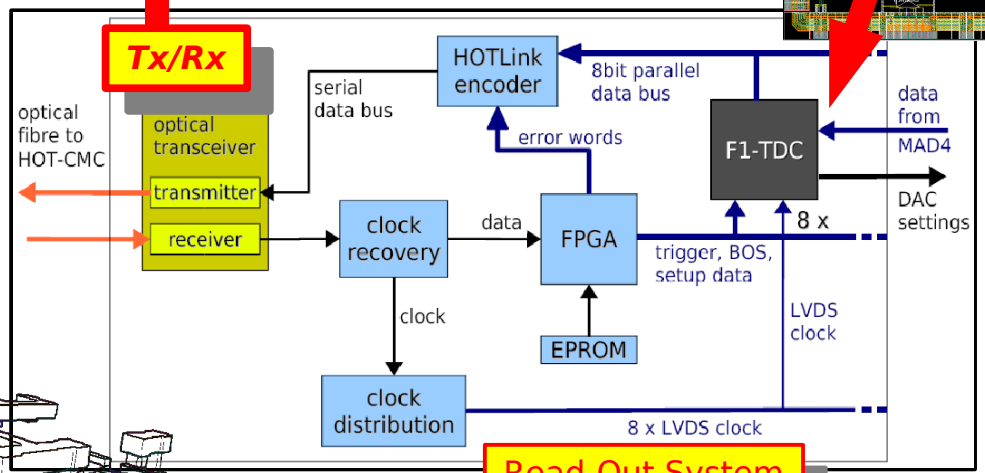
On/Off-Line Analysis



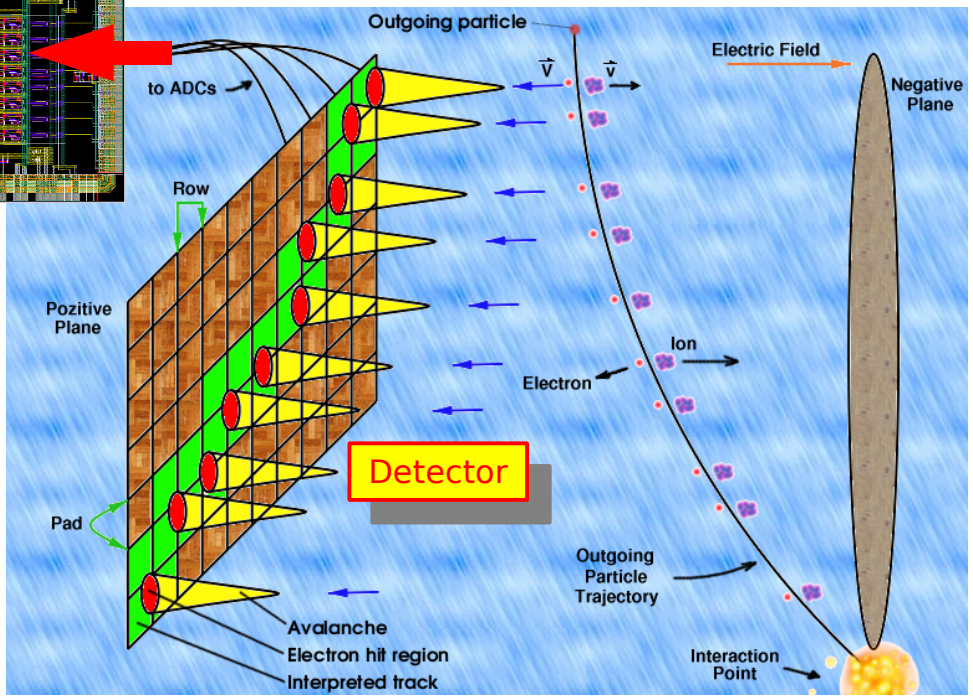
Front-End ASIC



Tx/Rx



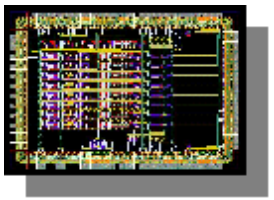
Read-Out System

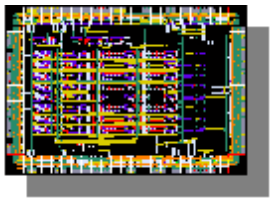


Content

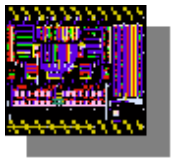
Developments I contributed to will be presented chronologically during my PhD with the emphasis of what is really produced and what publications they led.

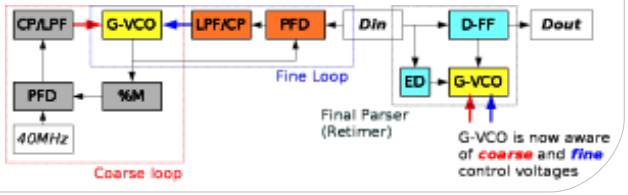
- **Development**
  - *The CMAD, a full custom front-end ASIC for the upgrade of the COMPASS RICH-I detector system @ CERN-SPS (complete)*
- **Publications**
  - *IEEE Conference records [x3]*
  - *Full NIM paper (in progress) [x1]*

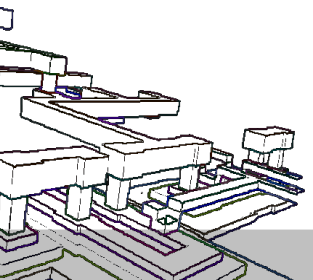




- **Development Target**
  - *CP-PLL based 4.8 Gb/s serializer and burst-mode capable Clock and Data Recovery (CDR) chip of Giga Bit Transceiver (GBT) for S-LHC @ CERN (in progress)*
- **Target Publications**
  - *IEEE Conference records (in progress) [x1]*
  - *IEEE Full paper [?]*

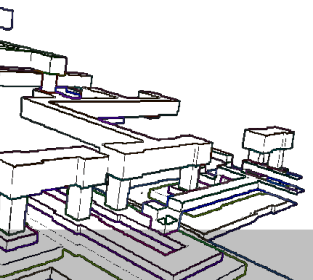






Outline

- Overview
  - Physics Interest
  - Introduction
  - Project Framework
- ASIC Design
  - Channel Architecture
  - Building Blocks
    - CSA, Shaper, BLH
    - 10-Bits D/A, LDO, band-gap
  - Simulations
- Measurements
- Conclusion



Physics Interest

- Spin content of nucleon:

$$\frac{1}{2} = \underbrace{\frac{1}{2}\Delta\Sigma}_{\sim 0.3} + \Delta G + L_q + L_g$$

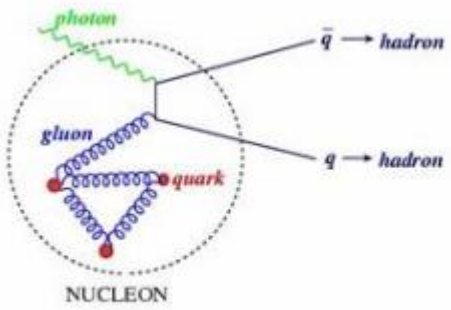
→ ***Probe:*** Photon Gluon Fusion (PGF)

→ ***Method #1:*** Open Charm Production

Spin-dependent asymmetry for charm-muo production

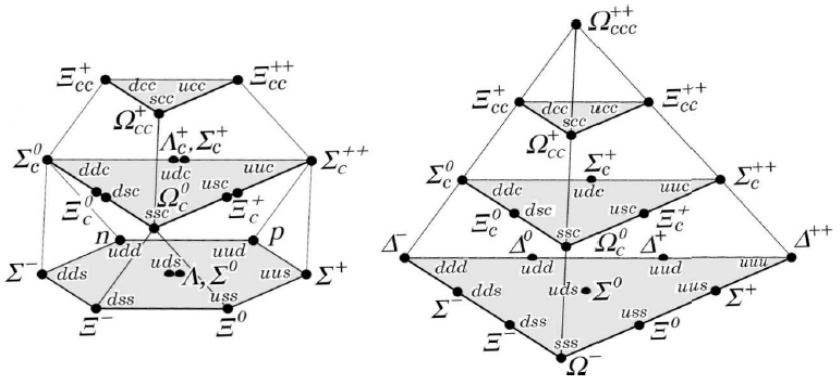
→ ***Method #2:*** High-p_T events

Opposite azimuth jets preserving the quark flavor and the direction of hard process



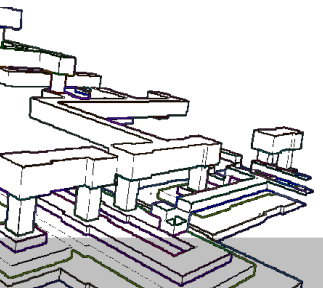
muon program

- Primakoff Reactions
  - EM effects on quarks
- Production and extraction of: $(q\bar{q})$ or (qqq)
  - Glue-Ball spectroscopy
  - Exotic states and exotic quantum numbers
  - Hexa/Penta/Tetra-Quarks
- Charmed and doubly charmed hadrons



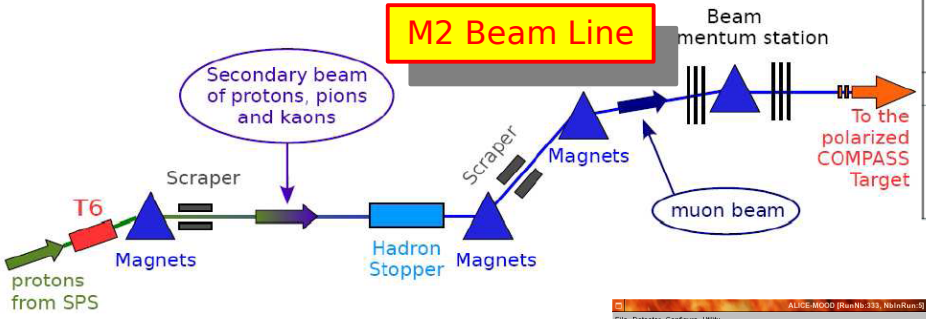
hadron program

	Muon Program	Hadron Program
Particles	μ^+	π, k, p
Energy (GeV/c)	$60 - 160$	100-300
Intensity (particle/spill)	$2 \cdot 10^8$	10^8
Beam size on targets (RMS in cm)	0.8	0.3 - 0.5

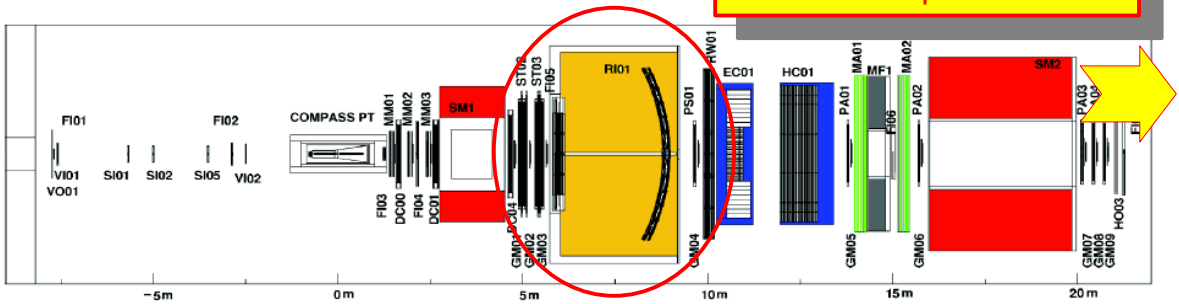


Introduction

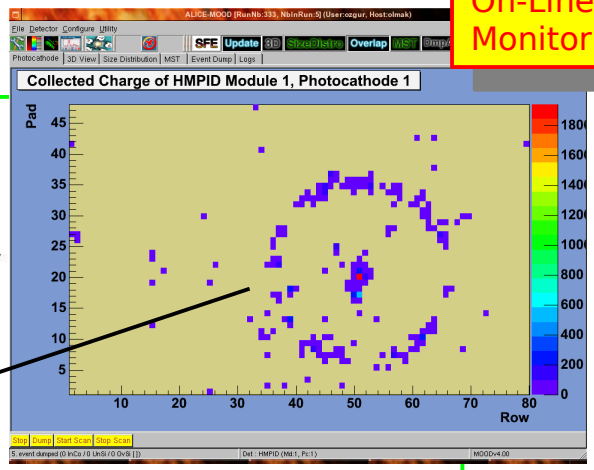
M2 Beam Line



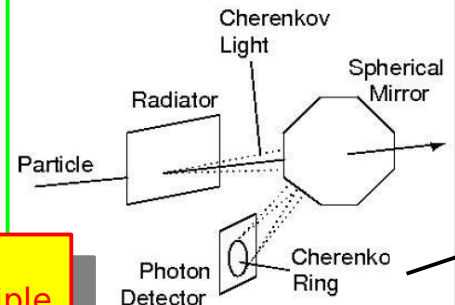
COMPASS Spectrometer



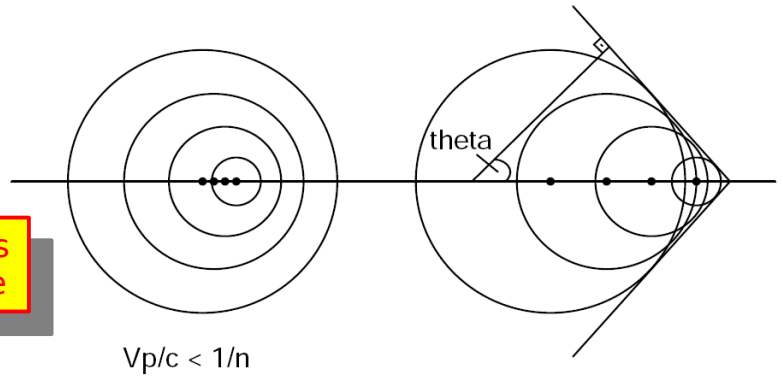
On-Line Monitor



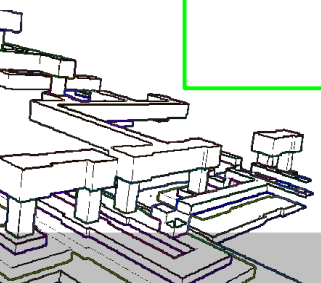
RICH Principle

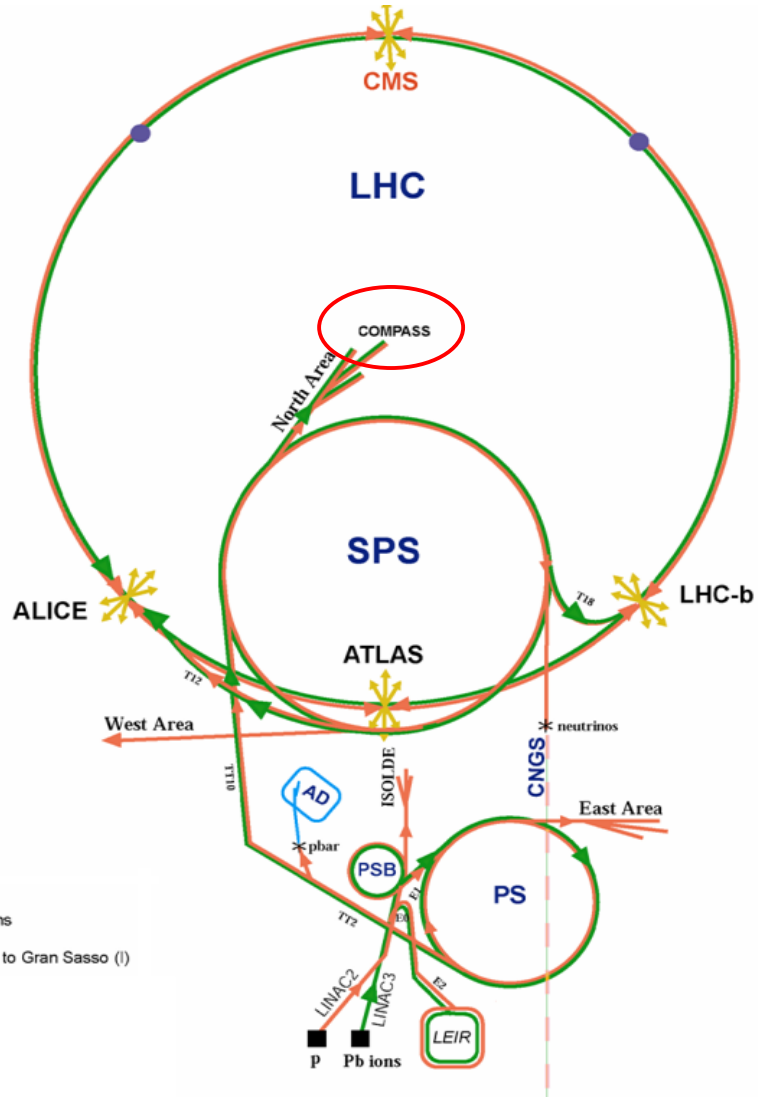


Huygens Principle



RICH Principle





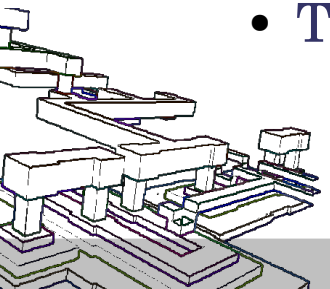
RICH-I Upgrade

- Previous ASIC developed for CMS, ***adopted*** for COMPASS RICH-I
  - FE stage ***not optimized***, un-necessary high gain
    - ***Noisy*** operation
    - ***Low threshold*** values can not be set
    - Low channel ***performance***
  - More ***flexibility*** needed

- Channel thresholds ***externally*** and ***globally*** set
  - Less room for ***channel equalization***
  - Need for ***external circuitry*** for reference generation

- Channel processing speed of ***5MHz/Ch*** must be sustained
  - Required by the new fast ***MPT***s

- Higher radiation tolerance
  - Technology change from 0.8µm BiCMOS to ***0.35µm CMOS***

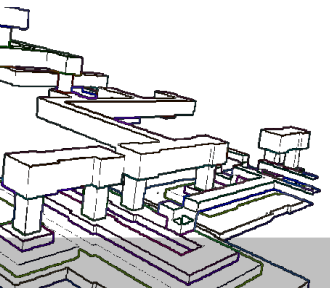


Project Framework

- Development of a new ***fast binary read-out ASIC*** for COMPASS at CERN
- Chip designed to replace an older ASIC (***the MAD-4***) for the ***RICH-I upgrade***
- User requirements :
  - Preserve the ***compatibility*** with the existing read-out
  - Reduced gain for ***MPT read-out***
  - Threshold/Baseline adjustable ***independently*** channel by channel
  - Hit rate > ***5 MHz/Channel***
  - Power consumption < ***30 mW/Channel***
- From 0.8µm BiCMOS to ***0.35µm CMOS***

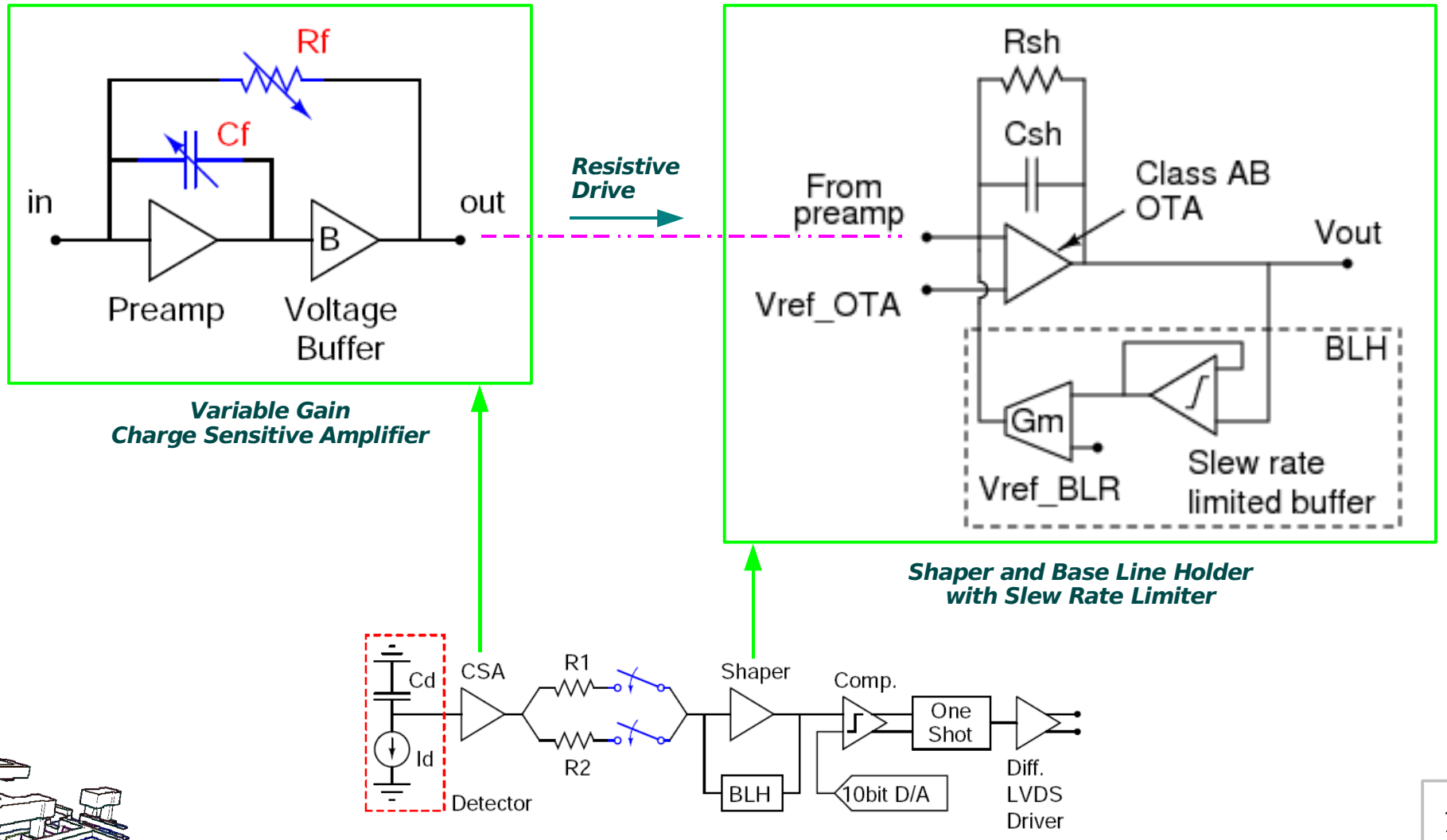
Technology	0.35 μm
Number of Channels	$8/Chip$
Preamplifier Gain Range	0.4-1.2 and 1.6-4.8 mV/fC
Preamplifier Gain Resolution	0.1 mV/fC
Peaking Time	10 ns
Hit Rate	>5 MHz/Ch
Chip Size	4.7x3.2 mm^2
Power (w/ LVDS Drivers)	26 mW

Specifications of The CMAD



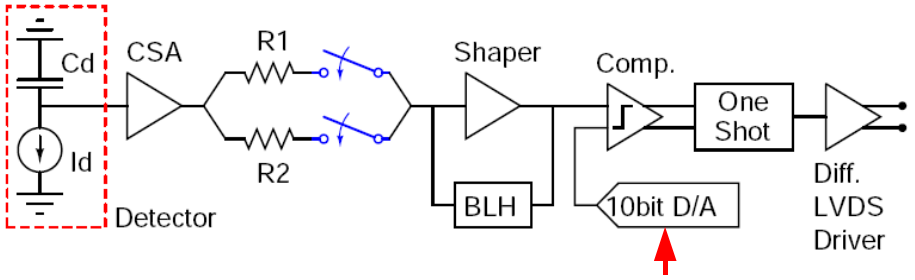
Channel Architecture

- Variable gain at Charge Sensitive Amplifier (CSA) stage by means of ***adjustable Cf and Rf***
- ***Rail-to-rail*** output Shaper (SH)
- Continuous-time Base Line Holder (BLH) with ***Slew Rate Limiter (SLR)***

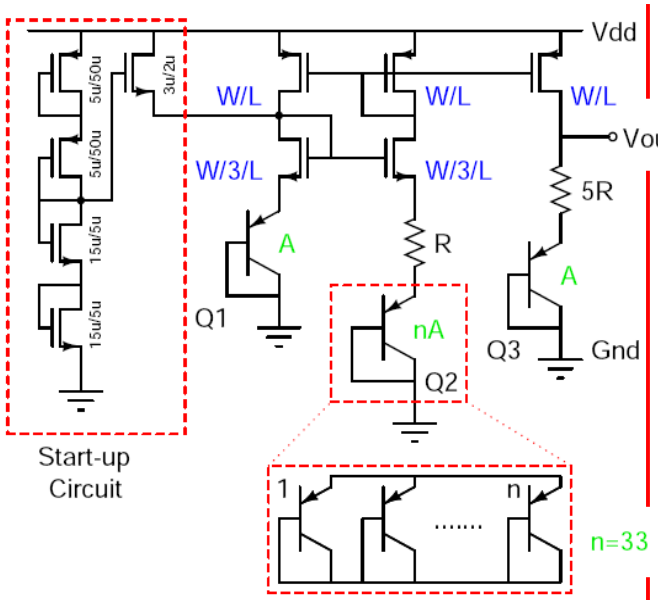


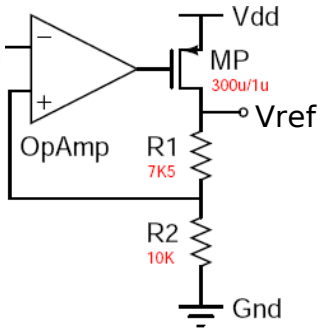
Blocks

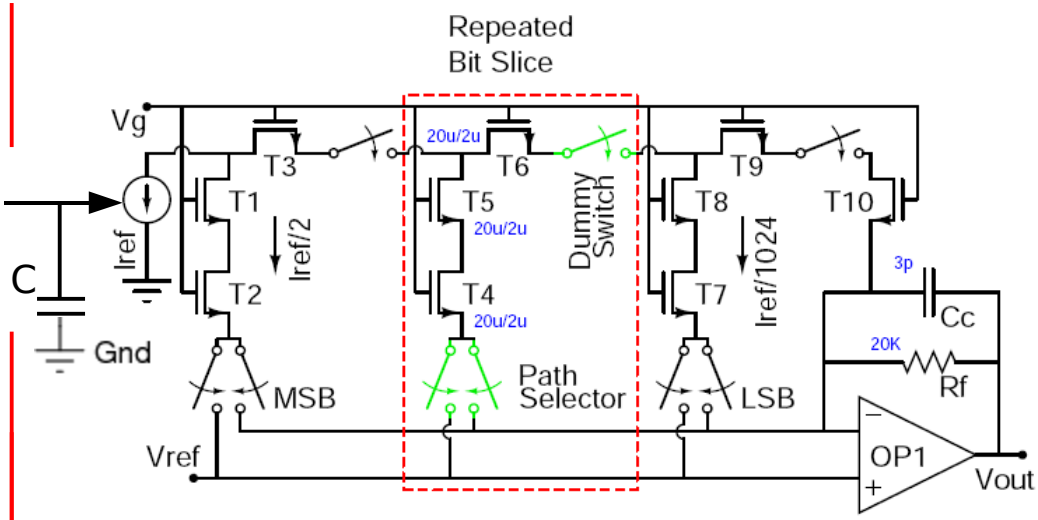
10-Bits D/A Converter and LDO



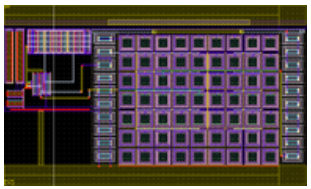
- ***Iref*** biases the circuit and is provided by an ***LDO*** (Low Drop Out regulator)
- LDO is referenced by a conventional ***on-chip band-gap*** reference voltage source

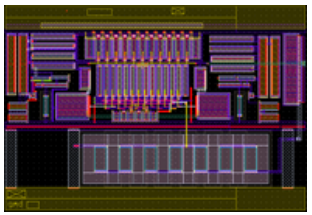


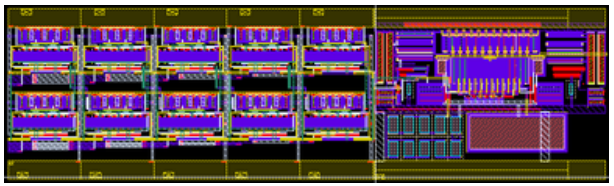


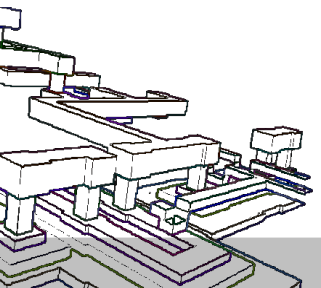


Band-gap ***LDO*** ***10-Bits transistor-only D/A***





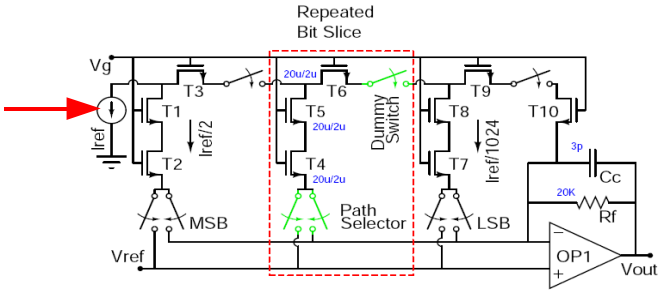


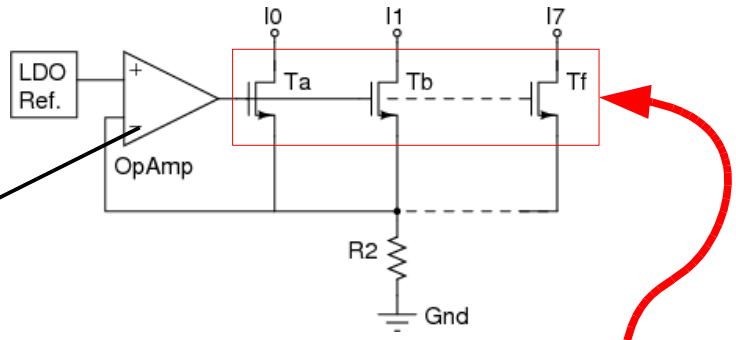


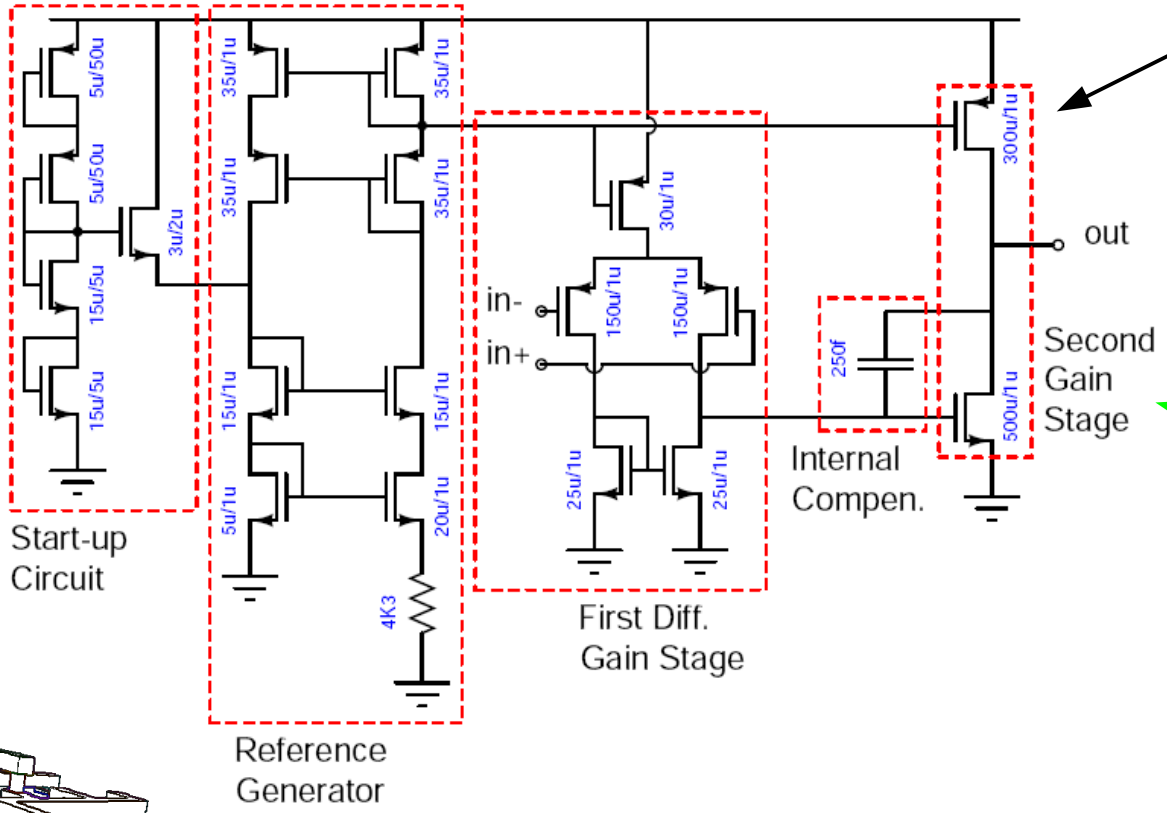
Blocks

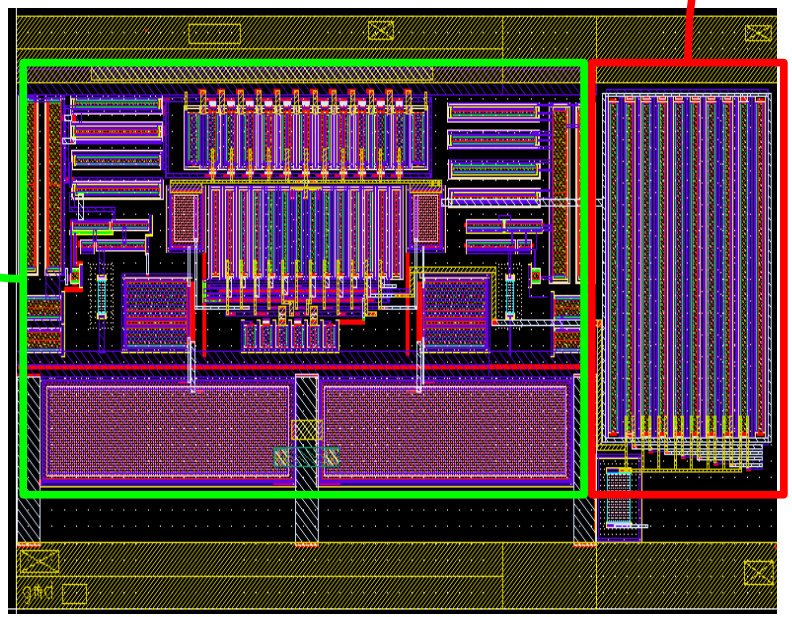
Current Sink

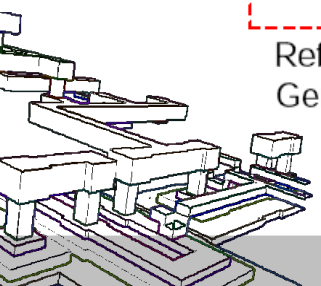
- Identical ***multi-channel*** architecture requiring identical biasing channels
- Rely on ***matching*** between :
  - Transistors *or*
  - Resistors
- MC showed : rely on transistor matching



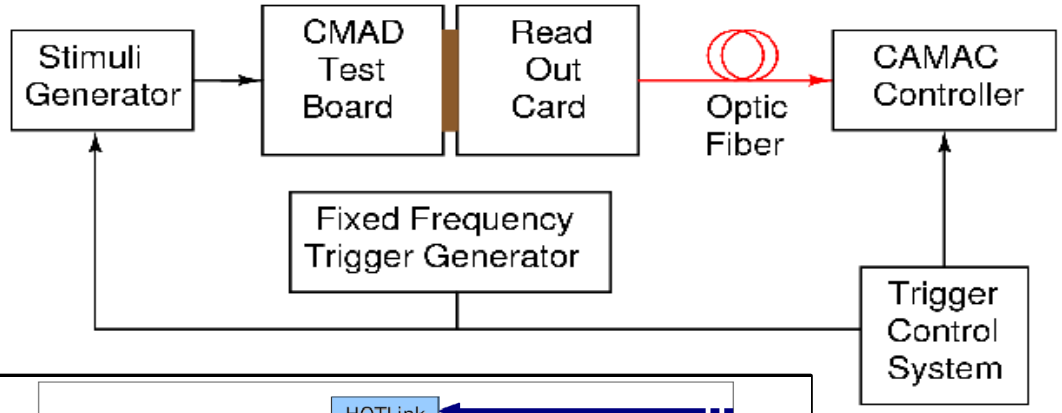


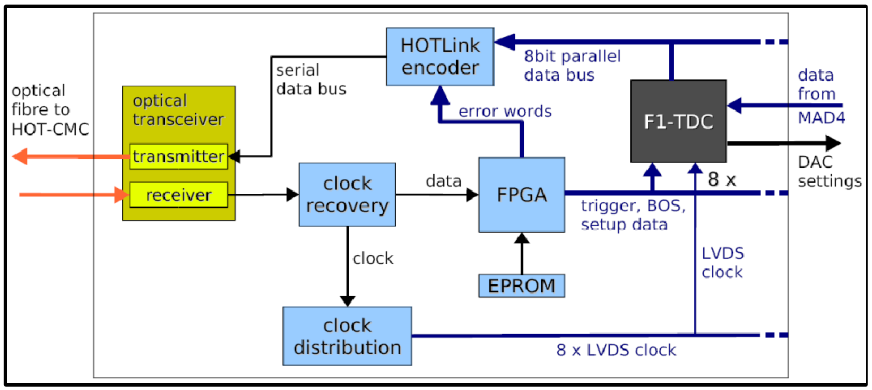


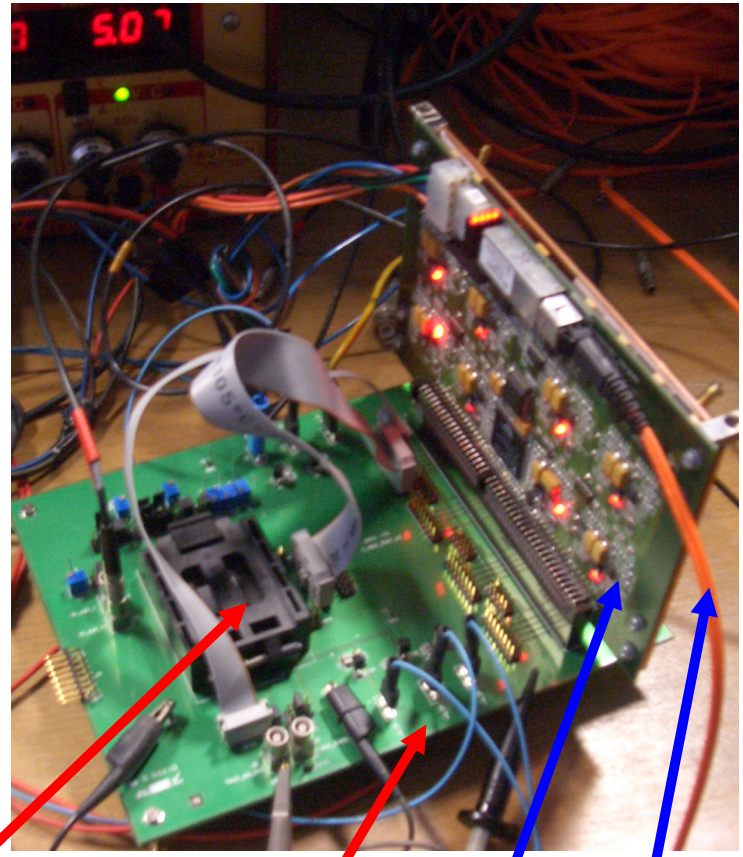




Test Setup



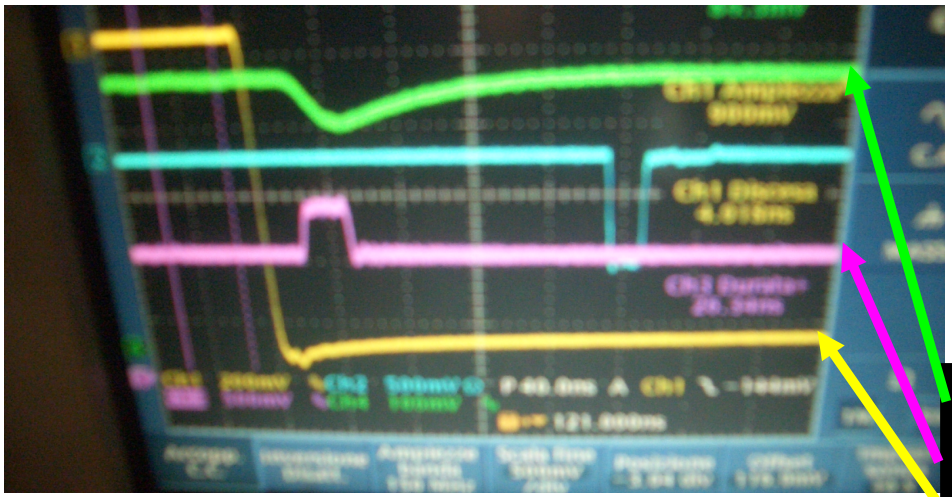




C-MAD and test board

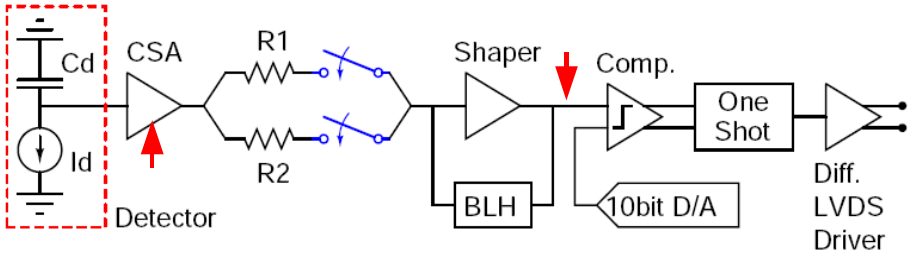
DREISAM Card

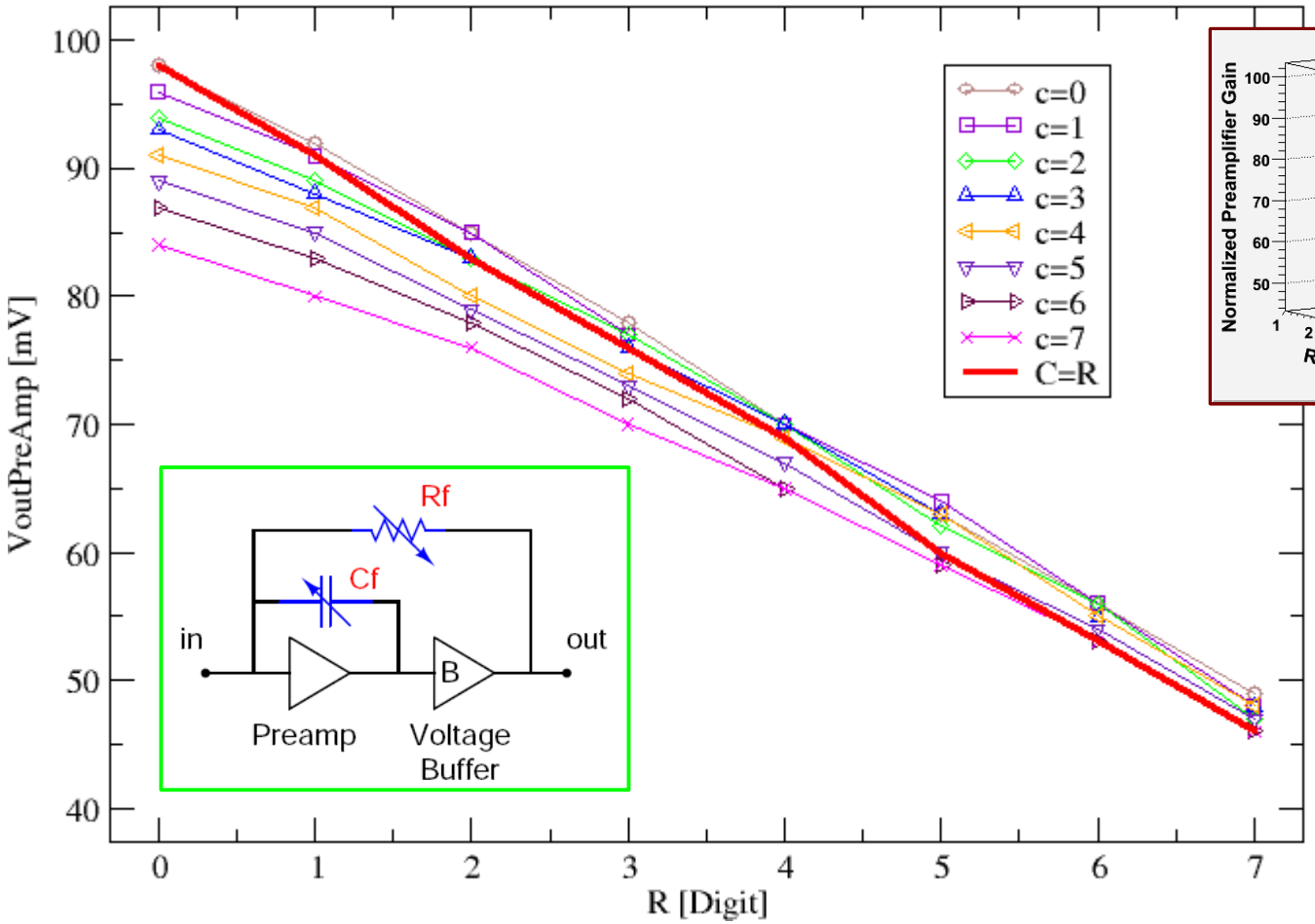
Optic fiber

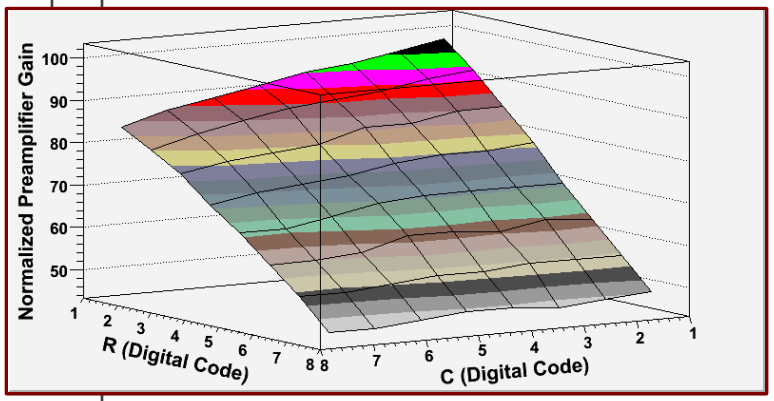


Pre-amplifier output
Comparator output
Input signal

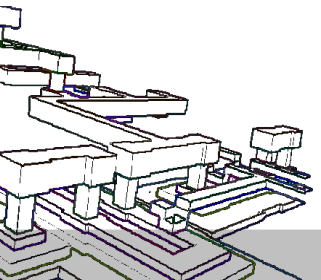
Variable Gain CSA Measurement Result





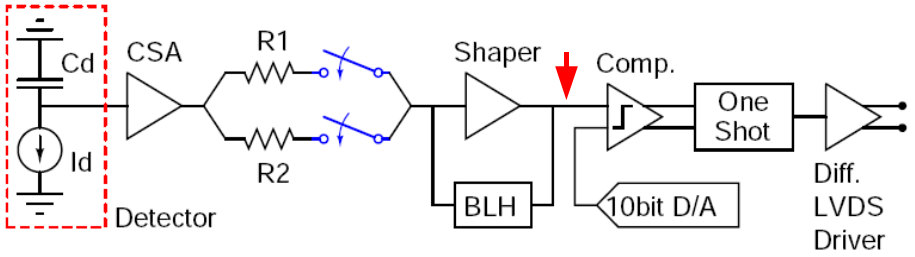


- ***Rf*** and ***Cf*** are adjusted accordingly to preserve the signal shape while changing the gain between ***0.4 mV/fC*** and ***1.2 mV/fC***.
- ***1.6mV/fC*** and ***4.8mV/fC*** in ***x4 mode***

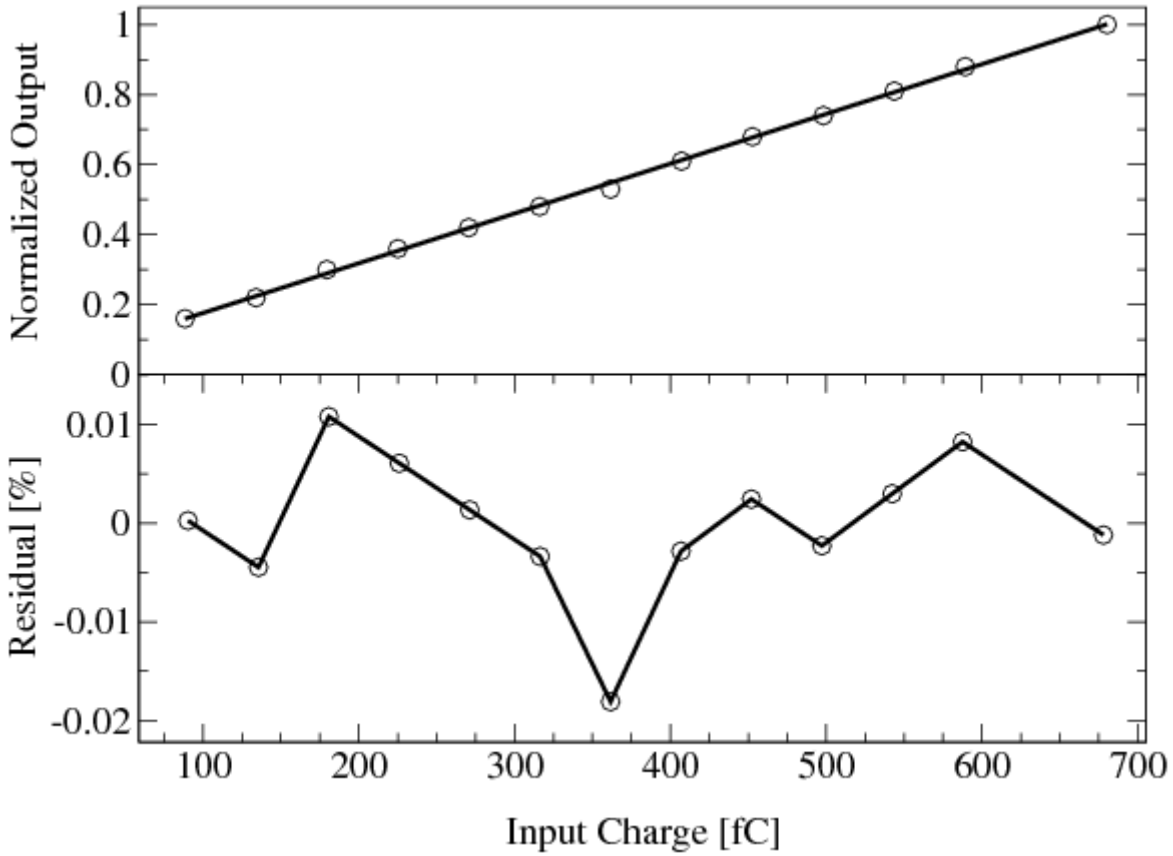


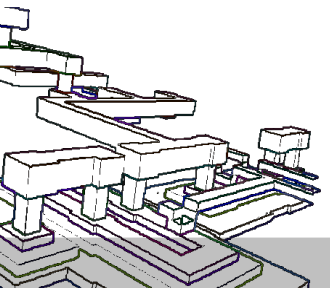
Gain Linearity

Measurement Result

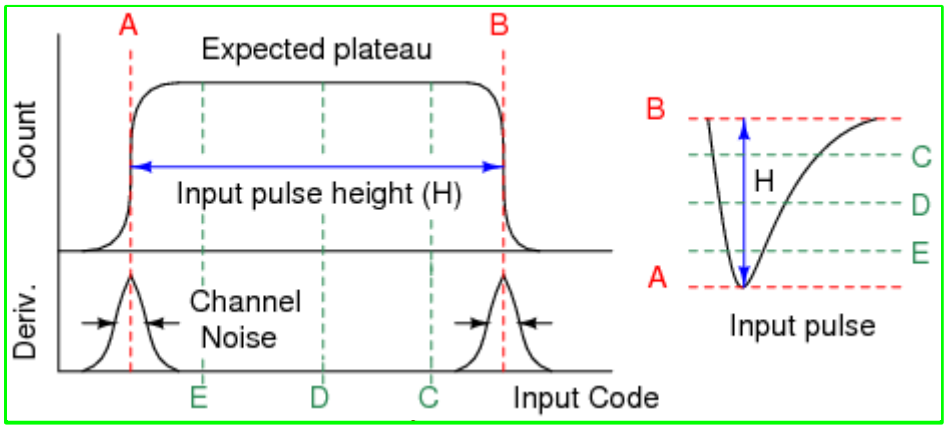


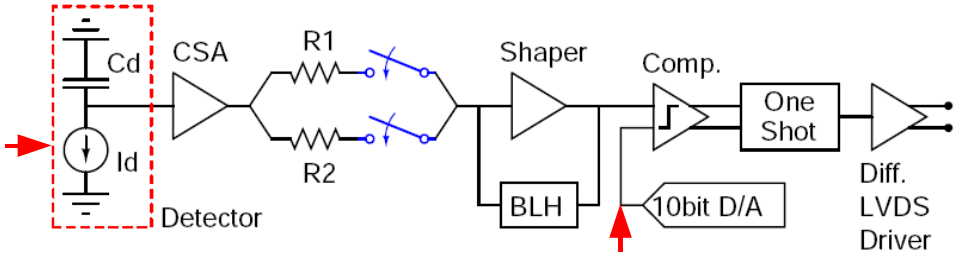
- Figure shows the measurement result and linear fit on the top and the residual on the bottom plot. Residual is less then ***2%***.





Threshold Scan





- A standard technique in nuclear and HEP experimental systems
- Channel equalization

Measured LED Signal

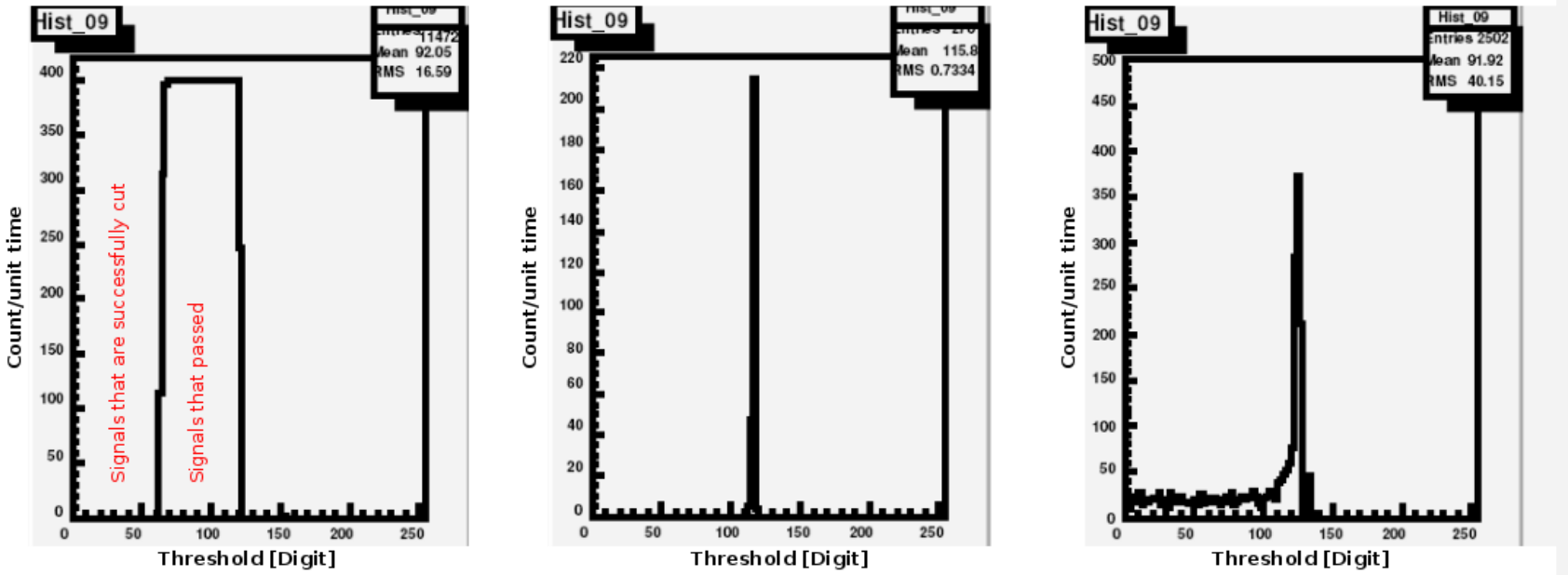
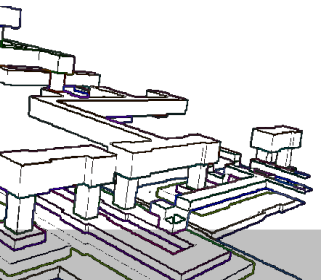
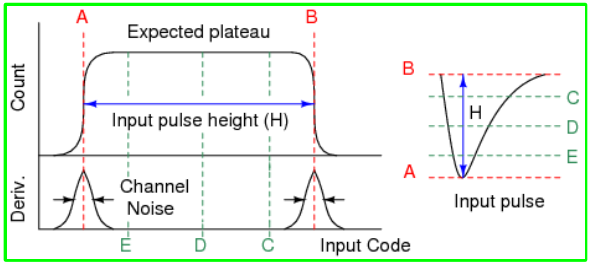


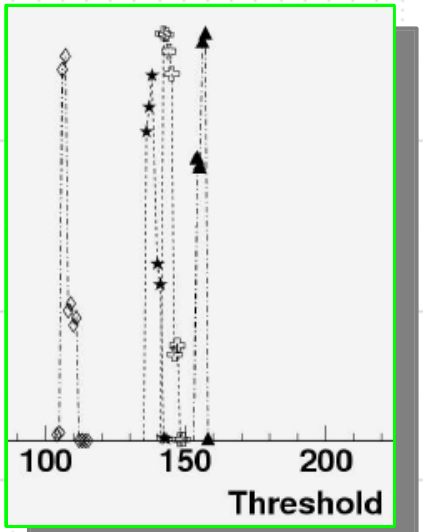
Figure : *Threshold scan test results, left figure shows the expected plateau, the middle figure shows the same distribution with a threshold very close to baseline (the peak is due to noise) and no input signal, and the right figure shows the results from the same measurement performed with PMTs and LEDs imitating Cerenkov radiation; there is not cut-signal-region, since the input signal was larger than the largest threshold value settable.*



Equalization

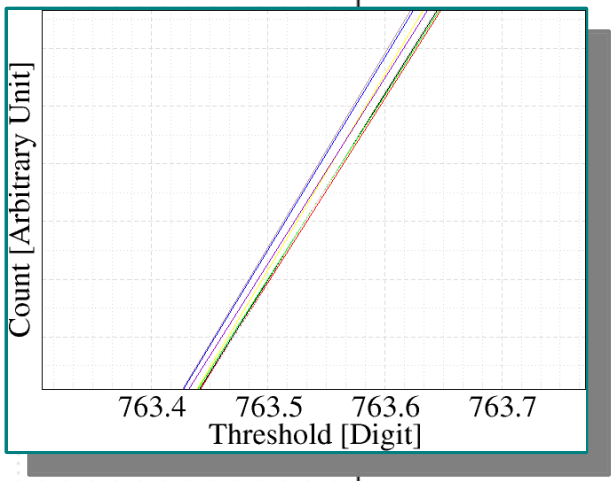
- Off-set between the ***8*** channels of the CMAD before and after channel equalization



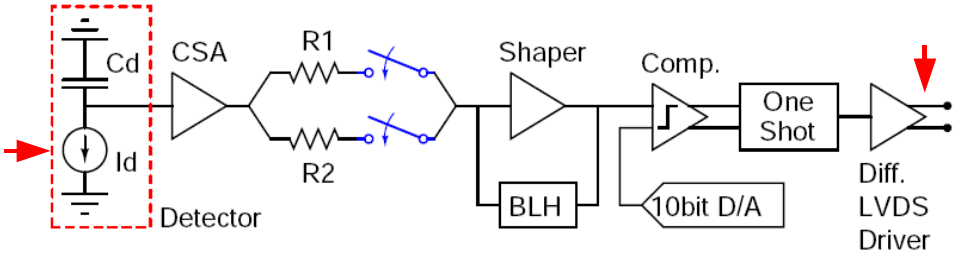


Before ***(50mV)***

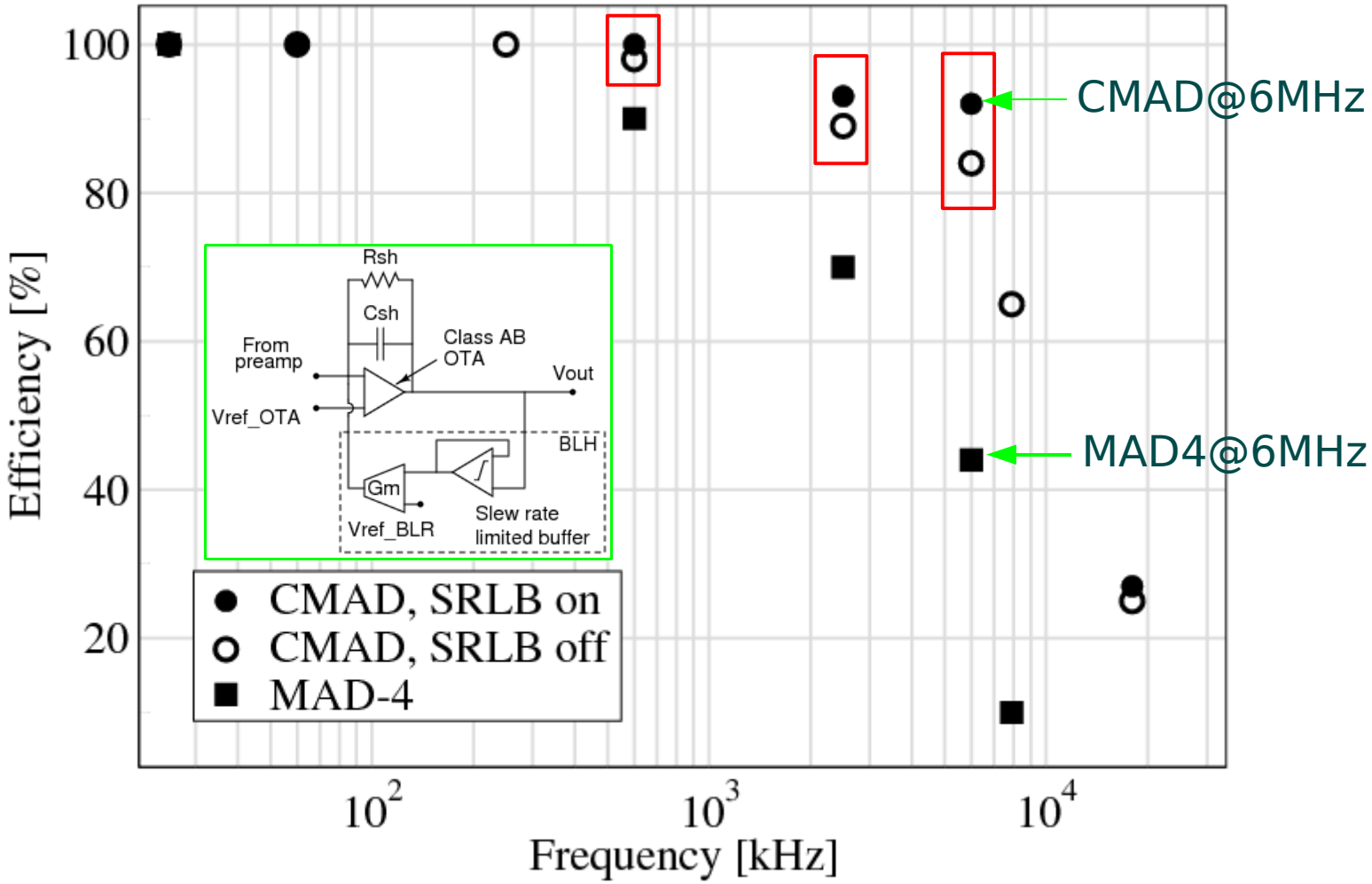
After ***(~0mV)***

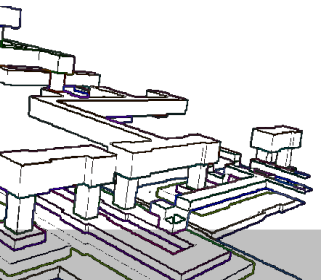


Efficiency



- Figure shows measurement results of
  the MAD-4,
  the CMAD w/o SRL and
  the CMAD w/ SRL

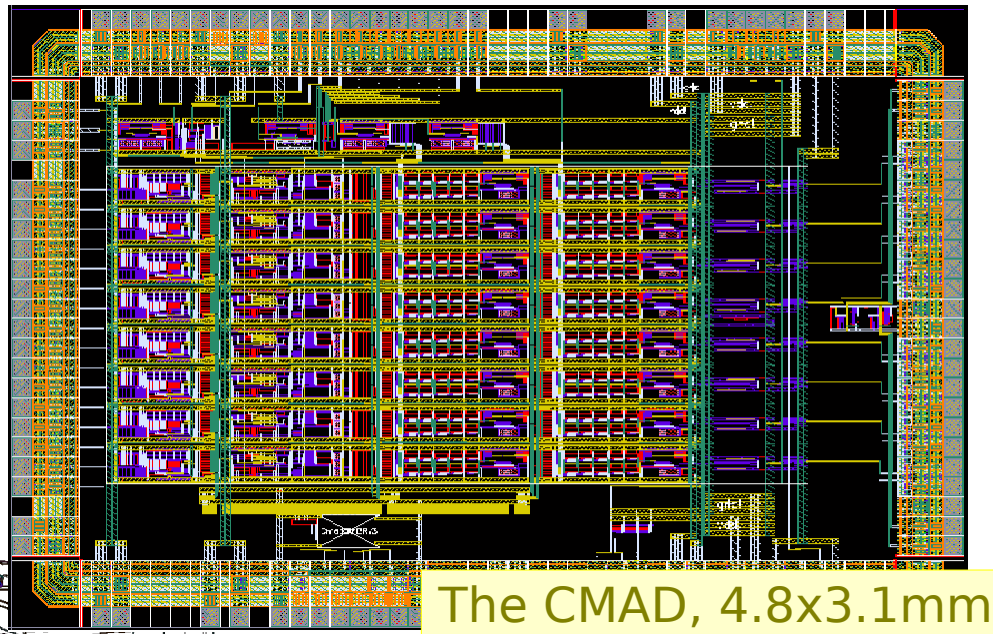




Conclusion

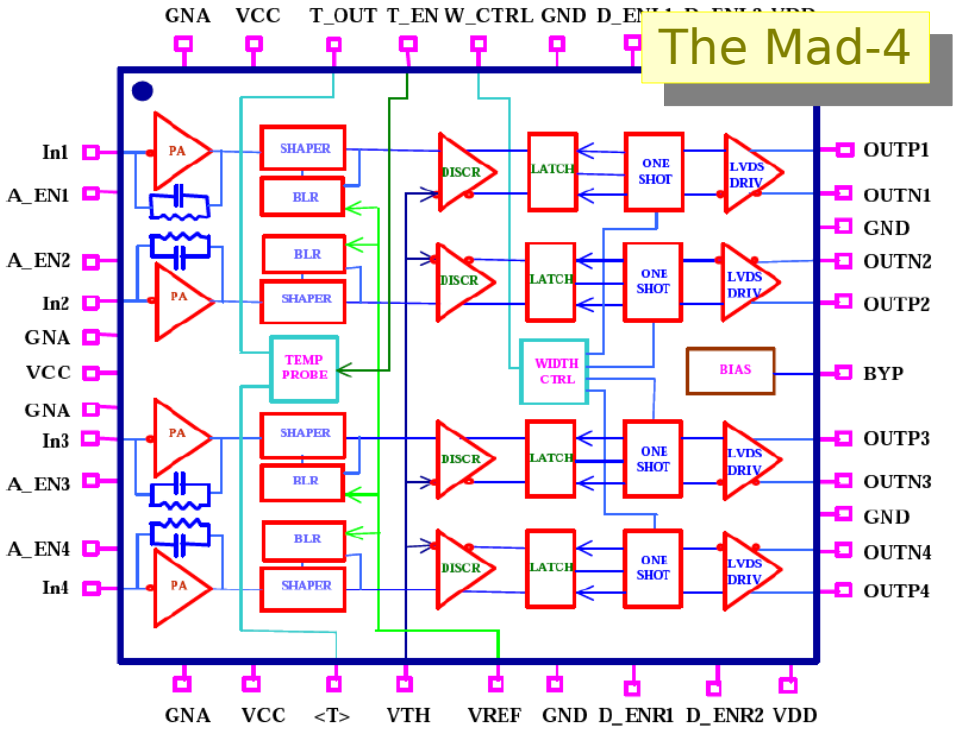
A ***new 8 channel***, ***full-custom ASIC***, named ***the CMAD***, designed for the readout of the ***RICH-I*** detector system of the ***COMPASS*** experiment at ***CERN*** is presented.

- In CMAD, ***threshold and baseline*** of each channel can be adjusted ***independent*** from the other channels.
- Thanks to ***low noise preamplifier***, lower thresholds can be set individually to improve front-end performance with a peaking time of ***10 ns.***
- Adjustable gain preamplifier from ***0.4 – 1.2 mV/fC and 1.6 – 4.8 mV/fC*** with steps of ***0.1 and 0.4 mV/fC***, respectively.
- The new design provides a ***higher speed*** of more than ***5 MHz/Ch*** with a ***power*** consumption of ***26 mW/Ch***.
- Noise level of ***1200 e⁻ @ 10pF*** input capacitance
- Layout (including the pad ring) size of ***4.8x3.1mm²***

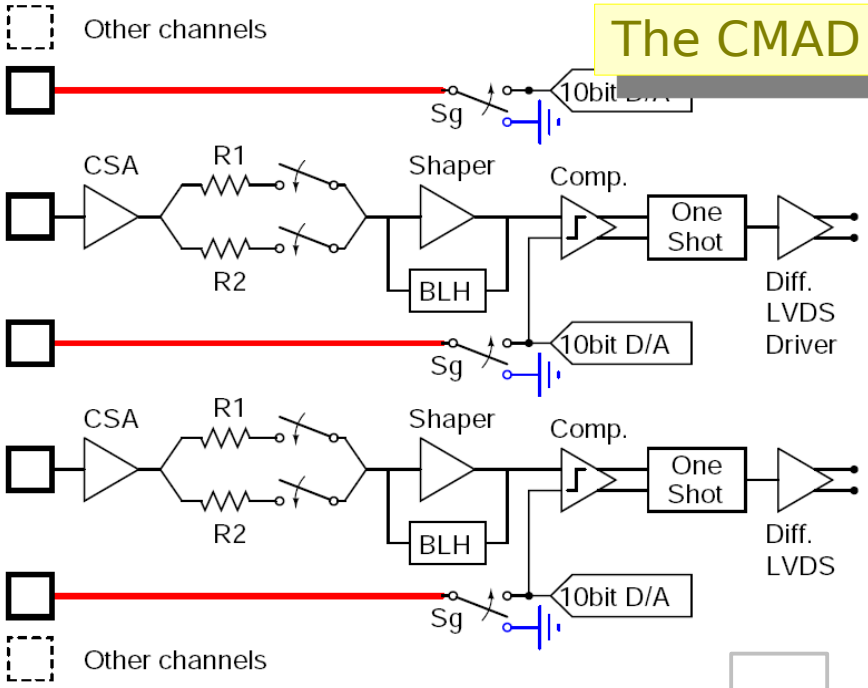


The CMAD, 4.8x3.1mm²

The Mad-4



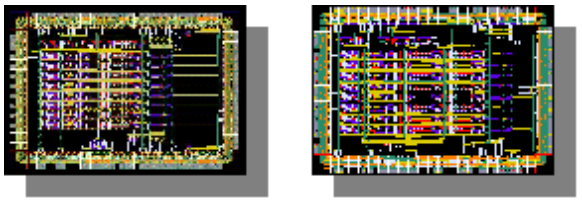
The CMAD



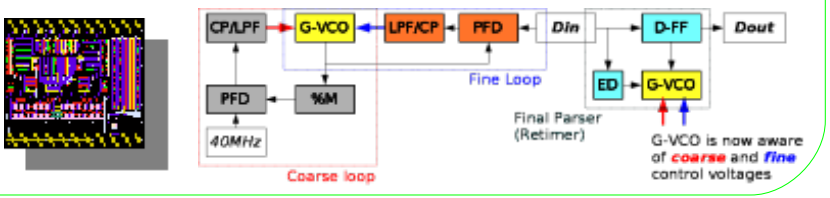
Content

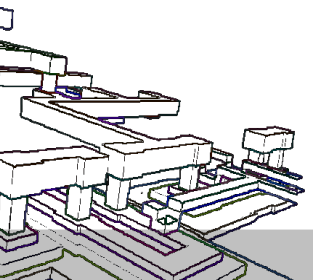
Developments I contributed to will be presented chronologically during my PhD with the emphasis of what is really produced and what publications they led.

- **Development**
  - *The CMAD, a full custom front-end ASIC for the upgrade of the COMPASS RICH-I detector system @ CERN-SPS (complete)*
- **Publications**
  - *IEEE Conference records [x3]*
  - *Full NIM paper (in progress) [x1]*



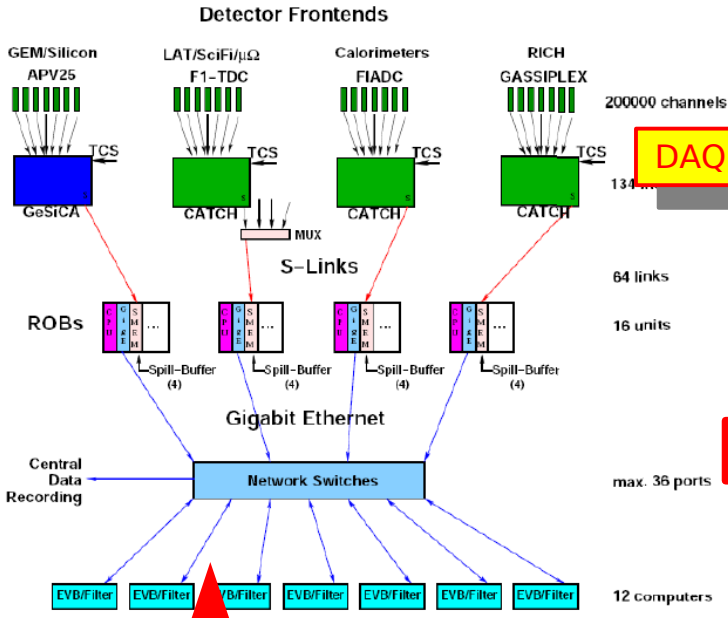
- **Development Target**
  - *CP-PLL based serializer at 4.8Gb/s and burst-mode capable Clock and Data Recovery (CDR) chip of Giga Bit Transceiver (GBT) for S-LHC @ CERN (in progress)*
- **Target Publications**
  - *IEEE Conference records (in progress) [x1]*
  - *IEEE Full paper [?]*





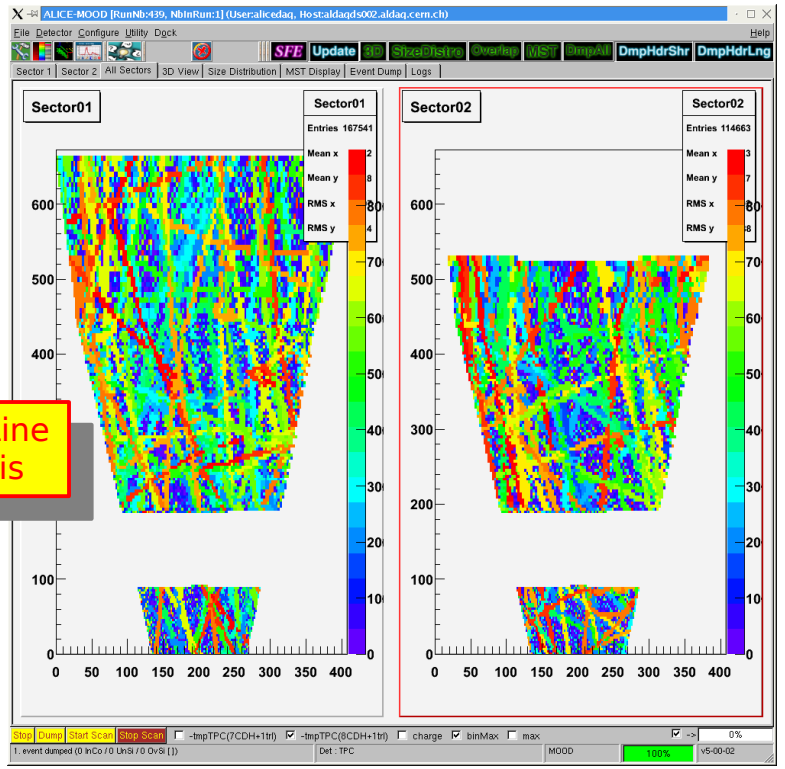
Terminology

- The ***Front-End*** ASIC sees the signal of detection first
- The ***Read-Out*** (RO) System "formats" this signal
- The ***Transceiver*** (Tx/Rx within RO) delivers the data to DAQ

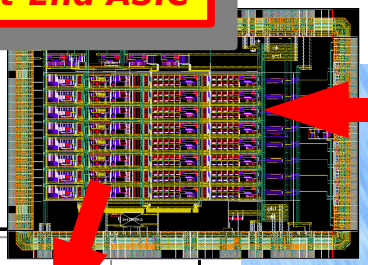


DAQ

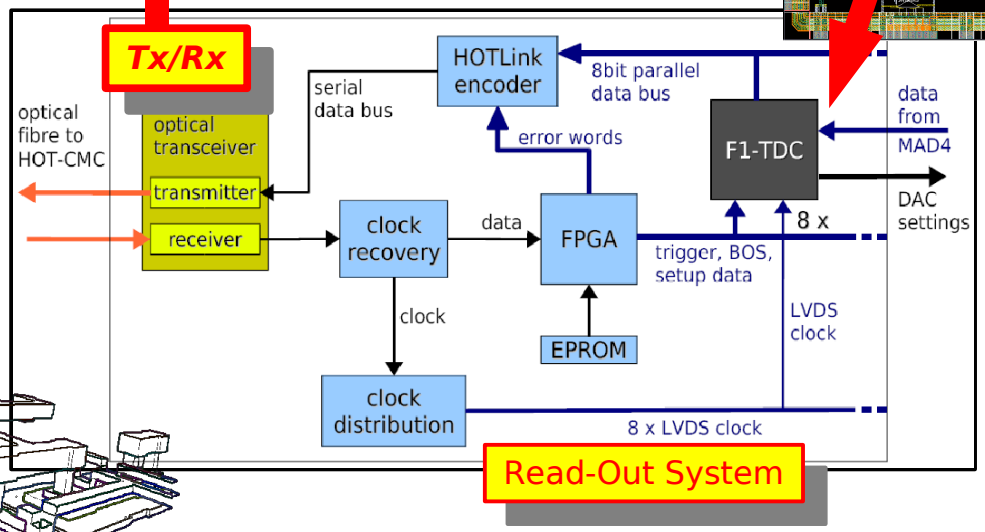
On/Off-Line Analysis



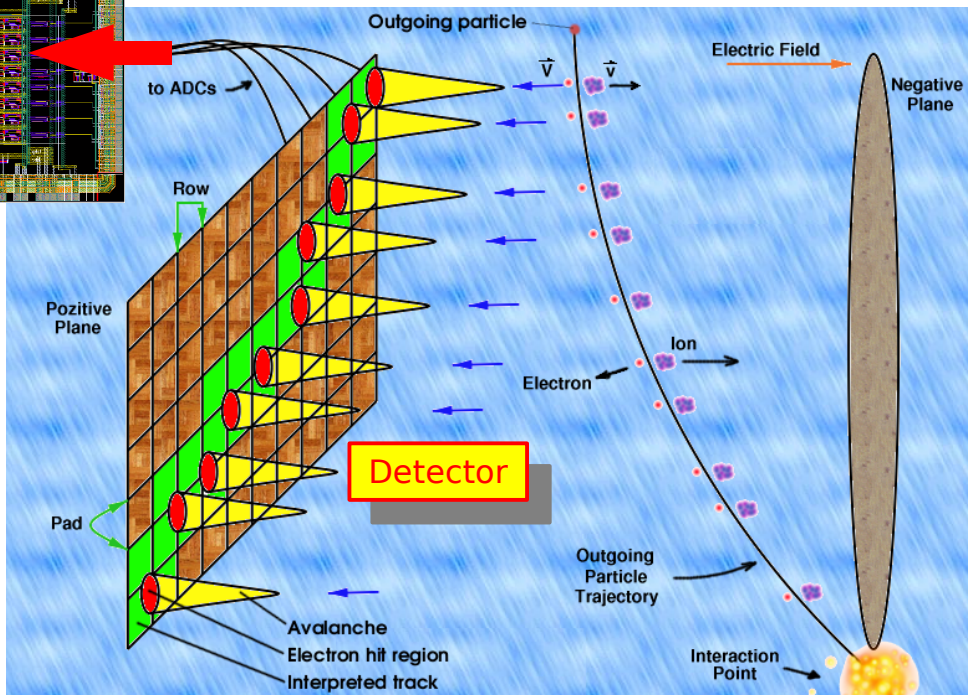
Front-End ASIC



Tx/Rx



Read-Out System



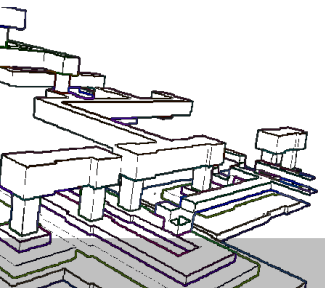
LHC Upgrade (1/2)

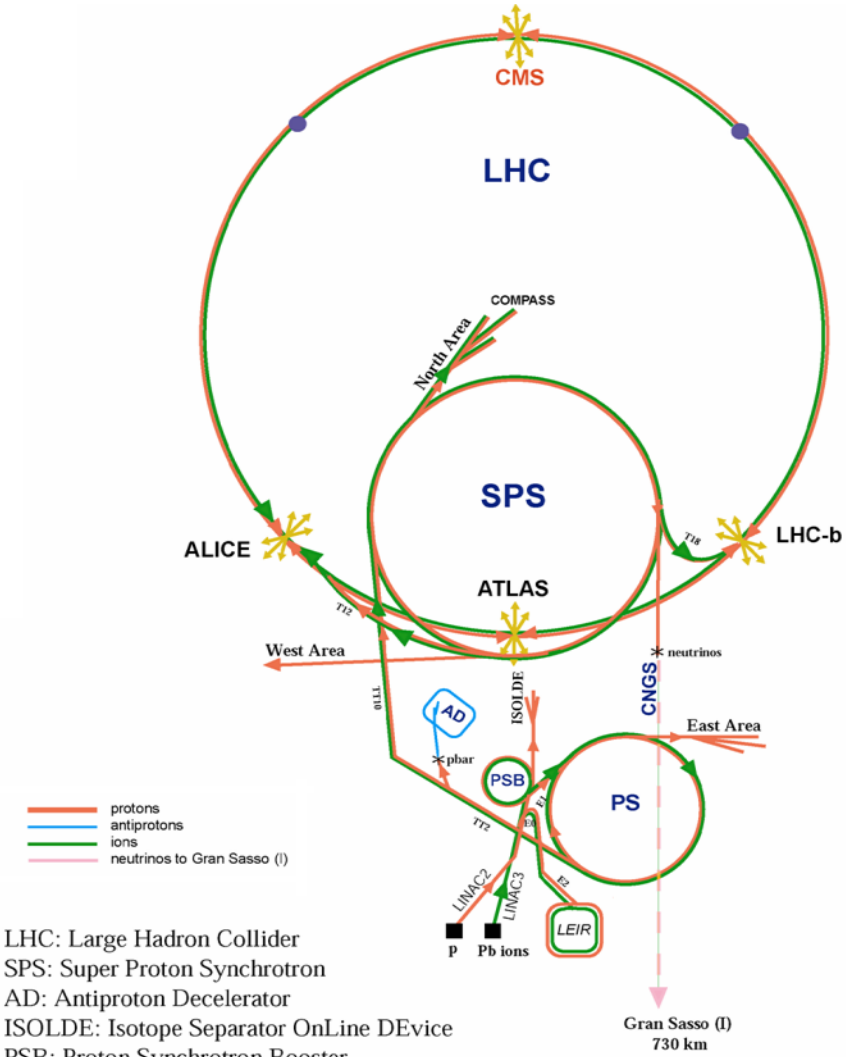
- 3 concurrent communication systems in HEP
  - ***SC*** – Slow Control
  - ***TTC*** – Timing Trigger and Control
  - ***DAQ*** – Data AcQuisition

- LHC Upgrade:
  - ***Electro-weak*** physics, ***Higgs*** physics, ***SUSY***, ***Extra dimensions***, Strongly coupled ***vector bosons*** (if no Higgs)
  - Half the bunch crossing interval (***12.5ns***)
  - ***Higher Luminosity*** – Detector Upgrades
  - Higher Energy – R&D needed for magnets (***>9T field***)

- GBT is proposed to be a 3-in-1 communication system replacing SC, TTC and DAQ together
  - ***Higher*** data rate @ SLHC
  - ***More*** error prone @ SLHC
  - Replacement needed due to ***rad-defects***
  - Lower maintenance ***cost***
  - Increased ***robustness***

In any case

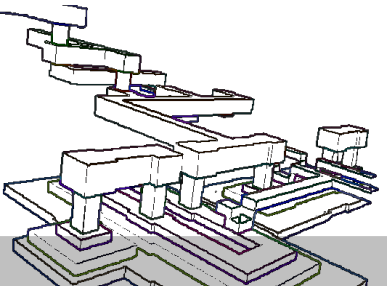




LHC: Large Hadron Collider
SPS: Super Proton Synchrotron
AD: Antiproton Decelerator
ISOLDE: Isotope Separator OnLine DEvice
PSB: Proton Synchrotron Booster
PS: Proton Synchrotron
LINAC: LINear ACcelerator
LEIR: Low Energy Ion Ring
CNGS: Cern Neutrinos to Gran Sasso

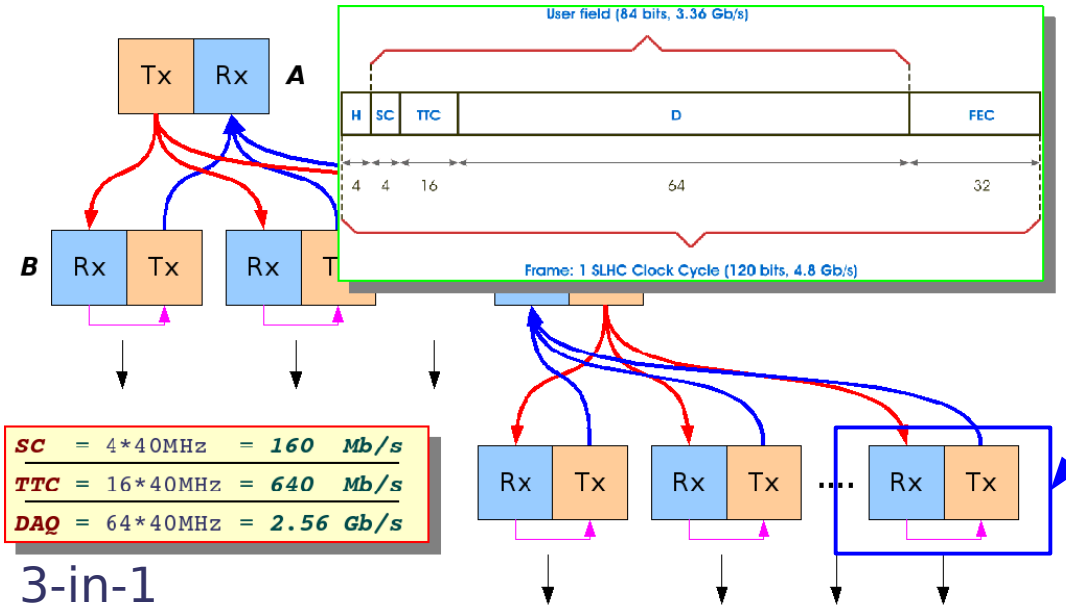
LHC Upgrade (2/2)

- ***1.*** Transmission of a ***single trigger type***,
- ***2. Several bunch crossing periods*** are required to transmit broadcast commands and slow control data,
- ***3.*** The system is ***unidirectional***. This required the late addition of an I2C network in order to control the TTCrx, necessitating the presence of an ***additional control path***,
- ***4.*** Although broadcast commands and slow control data are protected by ***error correction codes***, the trigger data are not,
- ***5.*** If not synchronized with the TTC signal source, the TTCrx generates a ***random clock frequency***. This is undesirable for purposes of system development and testing.



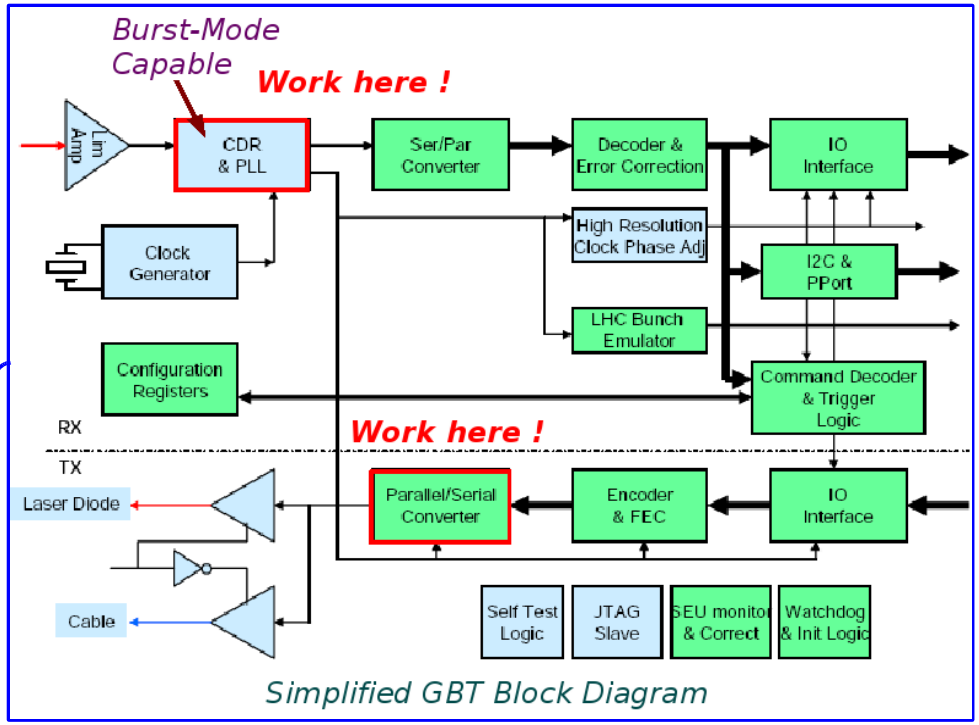
GBT-13

Line Format

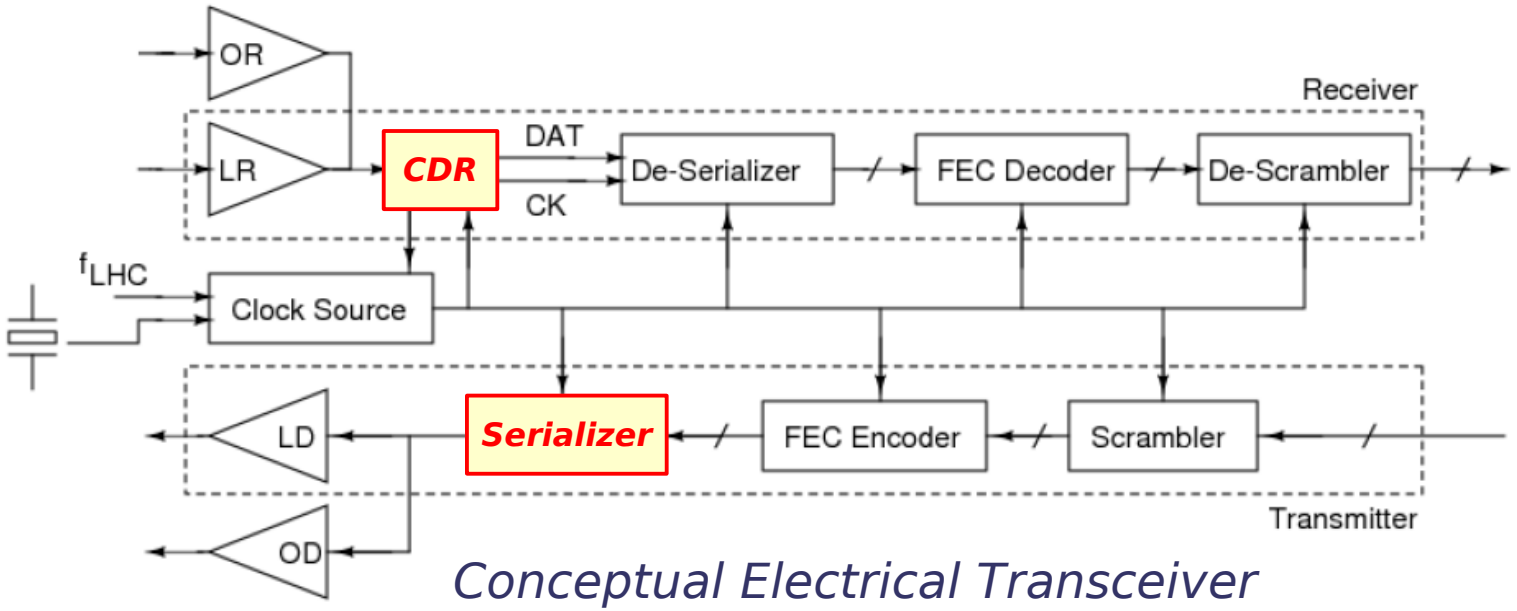


3-in-1

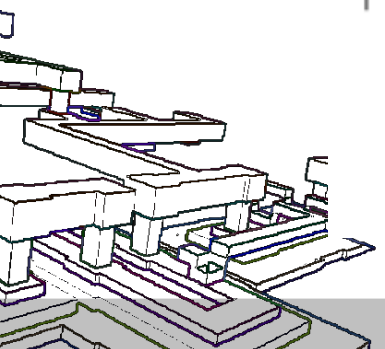
Transceiver GBT



Simplified GBT Block Diagram

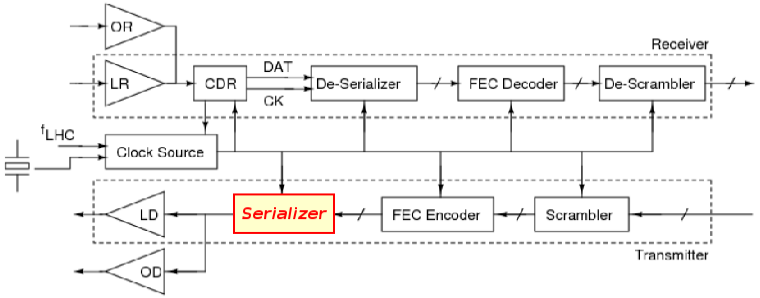


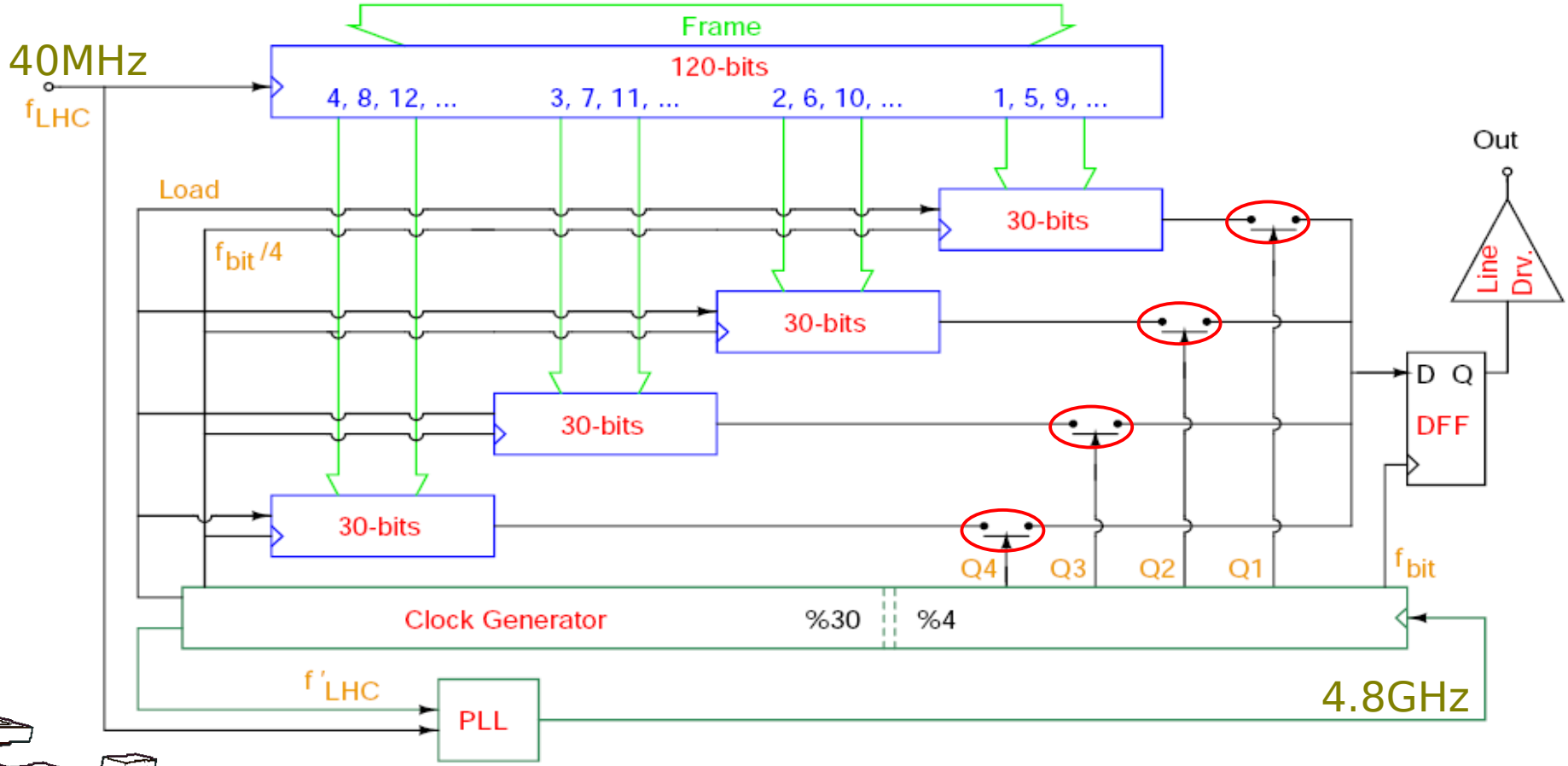
Conceptual Electrical Transceiver

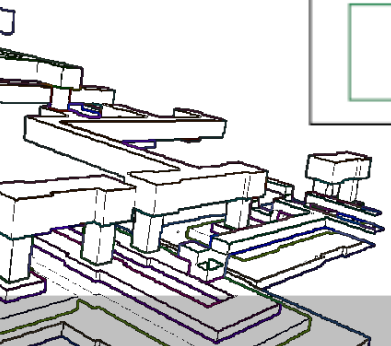


Serializer

- *CP-PLL based*
- *4.8 Gbit/s output speed*
- *40 MHz LHC clock as the reference for the CP-PLL*
- *Using only rising edges of clocks (no duty-cycle induced jitter)*
- *No MUX, 4 switches instead*
- *Triple redundant against SEU*
- *Output re-timer*

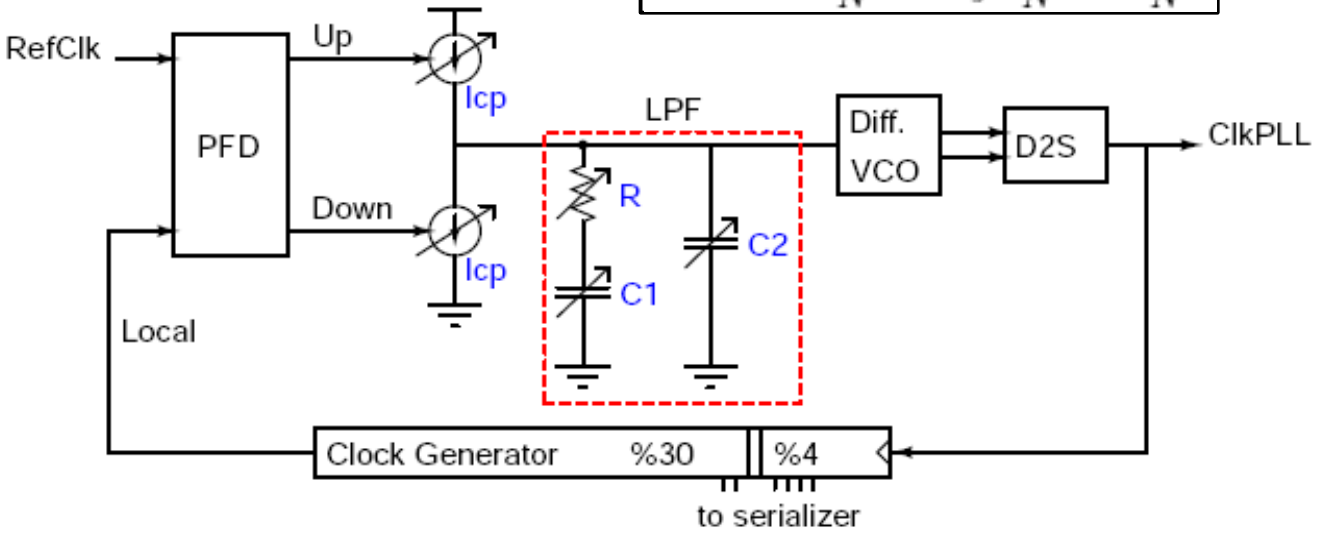


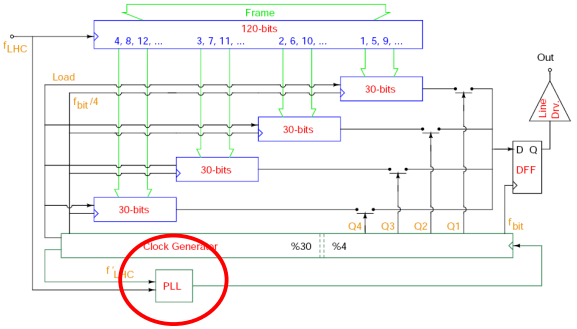




CP-PLL

$$T(s) = \frac{\omega_n{}^2(\tau s + 1)}{\frac{s^2}{N} + 2\xi s\frac{\omega_n}{N} + \frac{\omega_n{}^2}{N}}$$





$$K\tau < \frac{\omega_i \tau}{2\pi}$$

$$K\tau = \left[\frac{\pi}{\omega_i\tau}\left(1 + \frac{\pi}{\omega_i\tau}\right)\right]^{-1}$$

$$T_{ref2out}(s) = \frac{2\xi\omega_n N s + N\omega_n{}^2}{s^2 + 2\xi\omega_n s + \omega_n{}^2}$$

$$T_{vco2out}(s) = \frac{s^2}{s^2 + Ks + \frac{K_o I_p}{2\pi NC}}$$

$$\tau = R_2 C$$

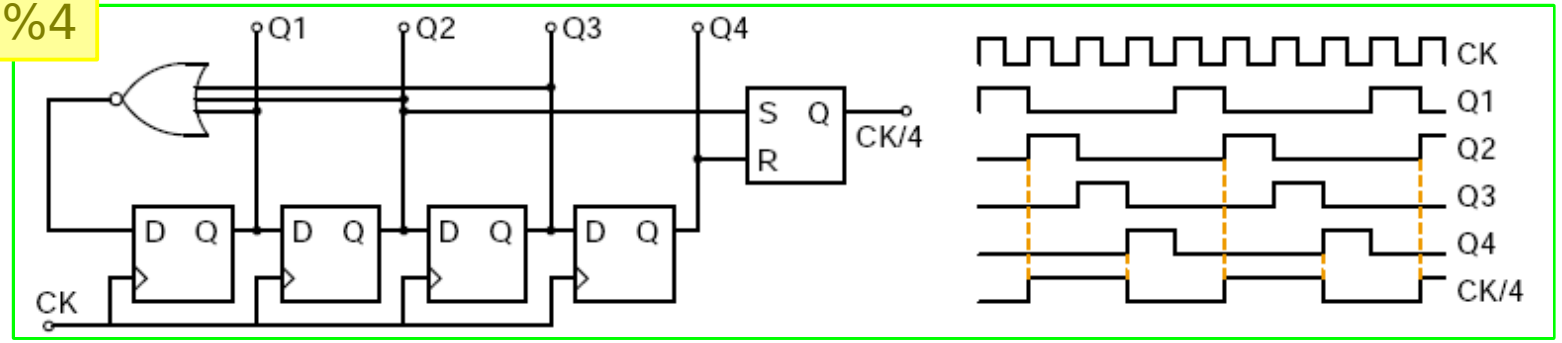
$$\omega_n = \sqrt{\frac{K_o I_p}{2\pi C N}}$$

$$\xi = \frac{\tau\omega_n}{2}$$

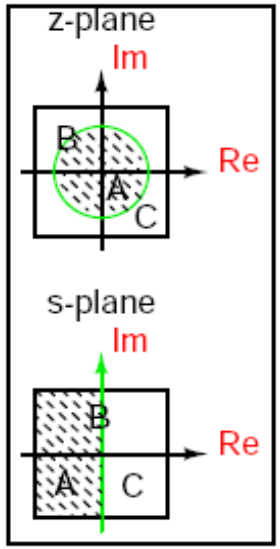
$$K = \frac{K_o I_p R_2}{2\pi N}$$

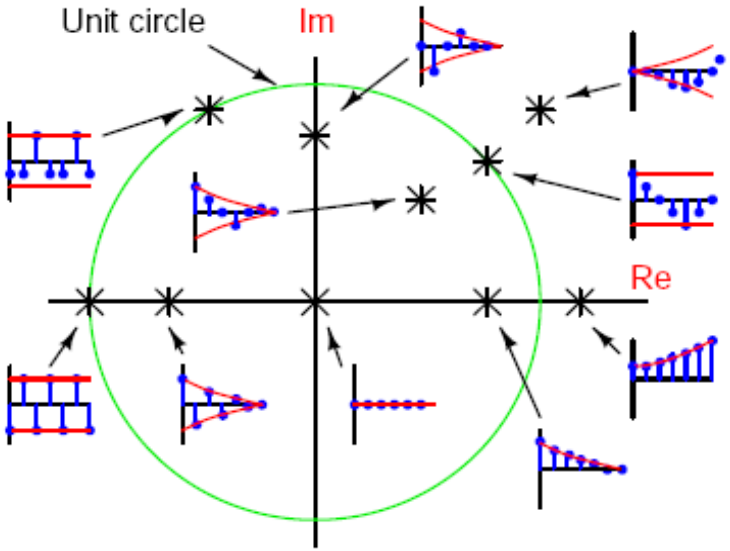
$$\theta_s = \frac{2\pi\Delta\omega}{K_o I_p R_s}$$

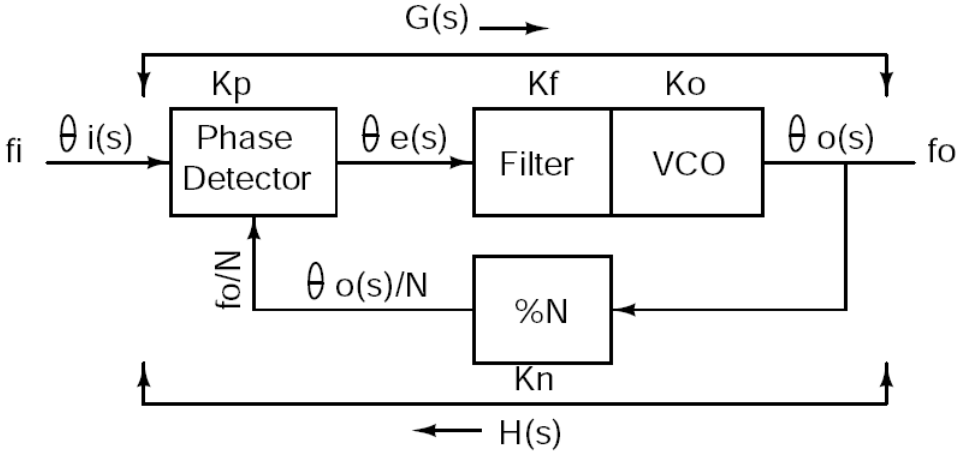
%4

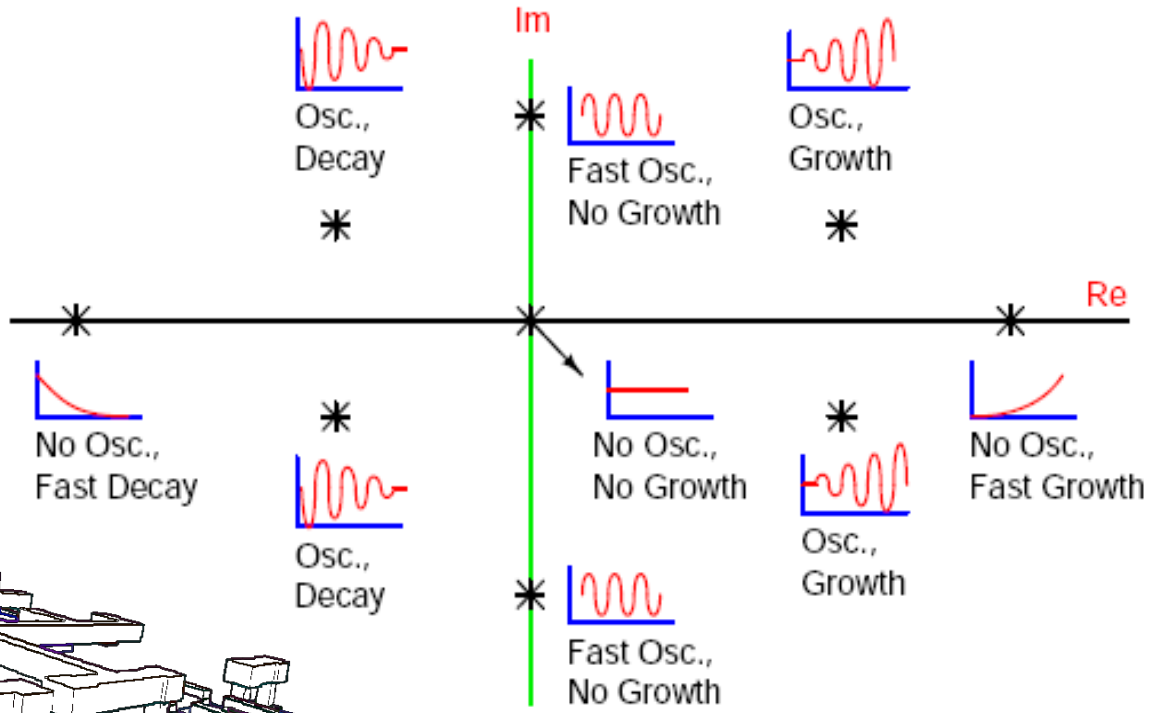


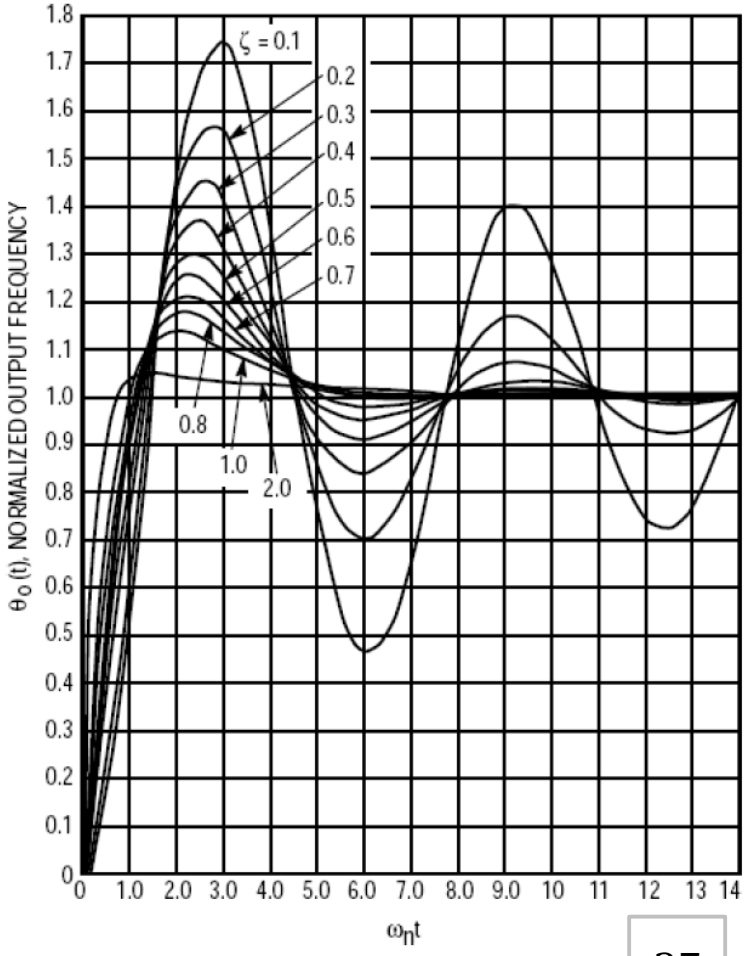
Feed-Back System Behavior

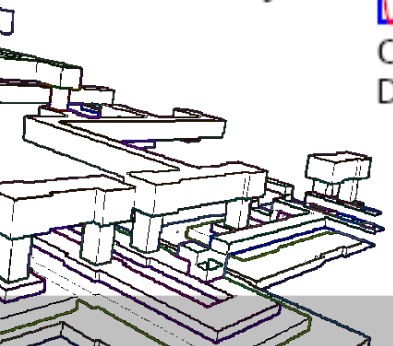












Cores #1 (CaPPeLLo : C/C++ and Octave m-files)

Core calculator function

CaPPeLLo.cxx - /home/oc/Documents/PhD/HDL/veril...

File Edit Search Preferences Shell Macro Windows Help

```
107             exit(1);
108         }
109     }
110 return 0;
111 }
112
113 //_____________________________________________________
114 int Calculate() {
115     C1=(Ko*Icp)/(2*PI*Wn*Wn*N);
116     C3max=0.1*C1;
117     C3min=0.02*C1;
118     C3=(C3min+C3max)/2.0;
119     Tau=(2*Ksi)/Wn;
120     R=Tau/C1;
121     Wz=2*PI*(1.0/(R*C1));
122     K=(Ko*Icp*R)/(2*PI*N);
123     Pmax=(2*PI*Wi*N)/(Ko*Icp);
124     Bl=((Wn/2.0)*(Ksi+1.0/(4.0*Ksi)));
125     proportionalTerm=Icp*R;
126     integralTerm=Icp/(C3*Wi);
127     if (R <= Rmax) isRmaxOk = true;
128     else isRmaxOk = false;
129     if (K*Tau < Wi*Tau) isStable = true;
130     else isStable = false;
131     return 0;
132 }
133
134 //_____________________________________________________
135 int Output() {
136     printf("Loop Parameters : \n\n");
137     printf("Wi in Mrad/s              : %f\n", (float)(Wi/1.0E6
138     printf("Wo in Grad/s              : %f\n", (float)(Wi*N*1.0
```

- ***CaPPeLLo.cxx*** is the code which corresponds to my ***hand calculations.***
- Run CaPPeLLo for each ***parameter set*** to have the required loop parameters for evaluating the ***closed loop behavior*** within ***Octave***

Octave m-file simulating the PLL for 16 parameter sets (*beginning*)

cppll_2.m - /home/oc/Documents/PhD/octave_workDir/

File Edit Search Preferences Shell Macro Windows Help

```
1  ##################################################################
2  #                                                                #
3  # OCTAVE script to evaluate the closed loop parameters of a CP-PLL. #
4  # Loop parameters are calculated by CaPPeLLo and "condition"     #
5  # variable corresponds to 16 different loop parameter sets.      #
6  #                                                                #
7  # http://www.ph.unito.it/~cobanogl,  Ozgur.Cobanoglu@cern.ch     #
8  #                                                                #
9  ##################################################################
10 #                                                                #
11 #       Kp        Icp      Tau        Kvco                      #
12 #      +-----+   +-----+   +-----+    +-----+                    #
13 # Wi -->| PFD |-->| CP |-->| LPF |--->| VCO |--+--> Wo #
14 #      +--+--+   +-----+   +-----+    +-----+  |                 #
15 #         |                                    |                 #
16 #         |  Wo/N          +----+     N*Wi     |                 #
17 #         +----------------| %N |<-------------+                 #
18 #                          +----+                                #
19 #                                                                #
20 # [System under consideration]                                   #
21 ##################################################################
22
23 clear
24
25 ########################################
26 # Parameter set for CP-PLL - by CaPPeLLo #  (Defterine bak)
27 ########################################
28
29 condition = 4;
30
31 if (condition == 0)
32 # CaPPeLLo -Kvco 35.0e9 -N 120 -Wi 40.0e6 -Ksi 4.67 -Wn 500.0e3  -Icp 5.0e-6
33 Wn      = 3.141593e6;
34 N       = 120.000000;
35 Ksi     = 4.670000;
36 Ko      = 219.911484e9;
37 Icp     = 5.000000e-6;
38 C       = 147.760040e-12;
39 R       = 20120.556641;
40 K       = 29342478.000000;
41 Tau     = 2.973014e-6;
42 KTau2   = 87.235606;
43 WiTau2  = 747.200016;
44 endif
45
46 if (condition == 1)
47 # CaPPeLLo -Kvco 35.0e9 -N 120 -Wi 40.0e6 -Ksi 4.67 -Wn 1000.0e3 -Icp 5.0e-6
48 Wn      = 6.283185e6;
49 N       = 120.000000;
50 Ksi     = 4.670000;
51 Ko      = 219.911484e9;
52 Icp     = 5.000000e-6;
53 C       = 36.940010e-12;
54 R       = 40241.113281;
55 K       = 58684956.000000;
```

Octave m-file simulating the PLL for 16 parameter sets (*end*)

cppll_2.m - /home/oc/Documents/PhD/octave_workDir/

File Edit Search Preferences Shell Macro Windows Help

```
271 ##############################
272 # Enter the transfer function #
273 ##############################
274
275 num = [(R*C*Wn^2) (Wn^2)];
276 den = [(1/N) (2*Ksi*Wn/N) (Wn*Wn/N)];
277 T = tf(num, den, 0, "ClkLHC", "ClkPLL/N");
278
279 # See what you entered
280
281 sysout(T)
282
283 # Extract some parameters to cross check
284
285 damp(T)
286
287 # Do I have a manageble system (Terminology from Control Theory)
288
289 is_observable (T)
290 is_controllable (T)
291 is_stabilizable (T)
292 is_detectable (T)
293 is_stable (T)
294
295 ######################
296 # Simulate the system #
297 ######################
298
299 wrange = logspace(log10(0.1),log10(10^10),100);
300 impulse(T, 1, 2*10^-6, 1000); figure;
301 step(T, 1, 2*10^-6, 1000);    figure;
302 bode(T, wrange);              figure;
303 rlocus(T, 0.001, 0.0, 1.0);                    figure;
304
305 ##################################
306 # Noise performance of the system #
307 ##################################
308
309 num = [(2*Ksi*Wn*N) (N*Wn^2)];
310 den = [1 (2*Ksi*Wn) (Wn^2)];
311 Tref2out = tf(num, den, 0, "LHC_Clock_Noise", "PLL_Output_Noise");
312 bode(Tref2out, wrange); figure;
313
314 sysout(T)
315 damp(T)
316
317 num = [1 0 0];
318 den = [1 (K) ((Ko*Icp)/(2*pi*N*C))];
319 Tvco2out = tf(num, den, 0, "VCO_Noise", "PLL_Output_Noise");
320 bode(Tvco2out, wrange); figure;
321
322 sysout(T)
323 damp(T)
324
325 pause
```

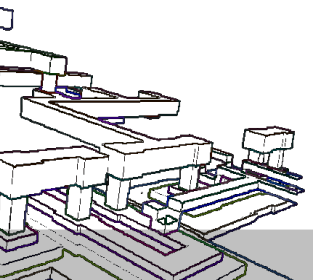
16 Parameter sets to be considered

conditions_2.txt - /home/oc/Documents/PhD/octave_workDir/

File Edit Search Preferences Shell Macro Windows Help

```
1  0   CaPPeLLo -Kvco 35.0e9 -N 120 -Wi 40.0e6 -Ksi 4.67 -Wn 500.0e3  -Icp 5.0e-6
2  1   CaPPeLLo -Kvco 35.0e9 -N 120 -Wi 40.0e6 -Ksi 4.67 -Wn 1000.0e3 -Icp 5.0e-6
3  2   CaPPeLLo -Kvco 35.0e9 -N 120 -Wi 40.0e6 -Ksi 4.67 -Wn 1500.0e3 -Icp 5.0e-6
4  3   CaPPeLLo -Kvco 35.0e9 -N 120 -Wi 40.0e6 -Ksi 4.67 -Wn 2000.0e3 -Icp 5.0e-6
5  4   CaPPeLLo -Kvco 35.0e9 -N 120 -Wi 40.0e6 -Ksi 4.67 -Wn 500.0e3  -Icp 10.0e-6
6  5   CaPPeLLo -Kvco 35.0e9 -N 120 -Wi 40.0e6 -Ksi 4.67 -Wn 1000.0e3 -Icp 10.0e-6
7  6   CaPPeLLo -Kvco 35.0e9 -N 120 -Wi 40.0e6 -Ksi 4.67 -Wn 1500.0e3 -Icp 10.0e-6
8  7   CaPPeLLo -Kvco 35.0e9 -N 120 -Wi 40.0e6 -Ksi 4.67 -Wn 2000.0e3 -Icp 10.0e-6
9  8   CaPPeLLo -Kvco 35.0e9 -N 120 -Wi 40.0e6 -Ksi 4.67 -Wn 500.0e3  -Icp 15.0e-6
10 9   CaPPeLLo -Kvco 35.0e9 -N 120 -Wi 40.0e6 -Ksi 4.67 -Wn 1000.0e3 -Icp 15.0e-6
11 10  CaPPeLLo -Kvco 35.0e9 -N 120 -Wi 40.0e6 -Ksi 4.67 -Wn 1500.0e3 -Icp 15.0e-6
12 11  CaPPeLLo -Kvco 35.0e9 -N 120 -Wi 40.0e6 -Ksi 4.67 -Wn 2000.0e3 -Icp 15.0e-6
13 12  CaPPeLLo -Kvco 35.0e9 -N 120 -Wi 40.0e6 -Ksi 4.67 -Wn 500.0e3  -Icp 20.0e-6
14 13  CaPPeLLo -Kvco 35.0e9 -N 120 -Wi 40.0e6 -Ksi 4.67 -Wn 1000.0e3 -Icp 20.0e-6
15 14  CaPPeLLo -Kvco 35.0e9 -N 120 -Wi 40.0e6 -Ksi 4.67 -Wn 1500.0e3 -Icp 20.0e-6
16 15  CaPPeLLo -Kvco 35.0e9 -N 120 -Wi 40.0e6 -Ksi 4.67 -Wn 2000.0e3 -Icp 20.0e-6
```

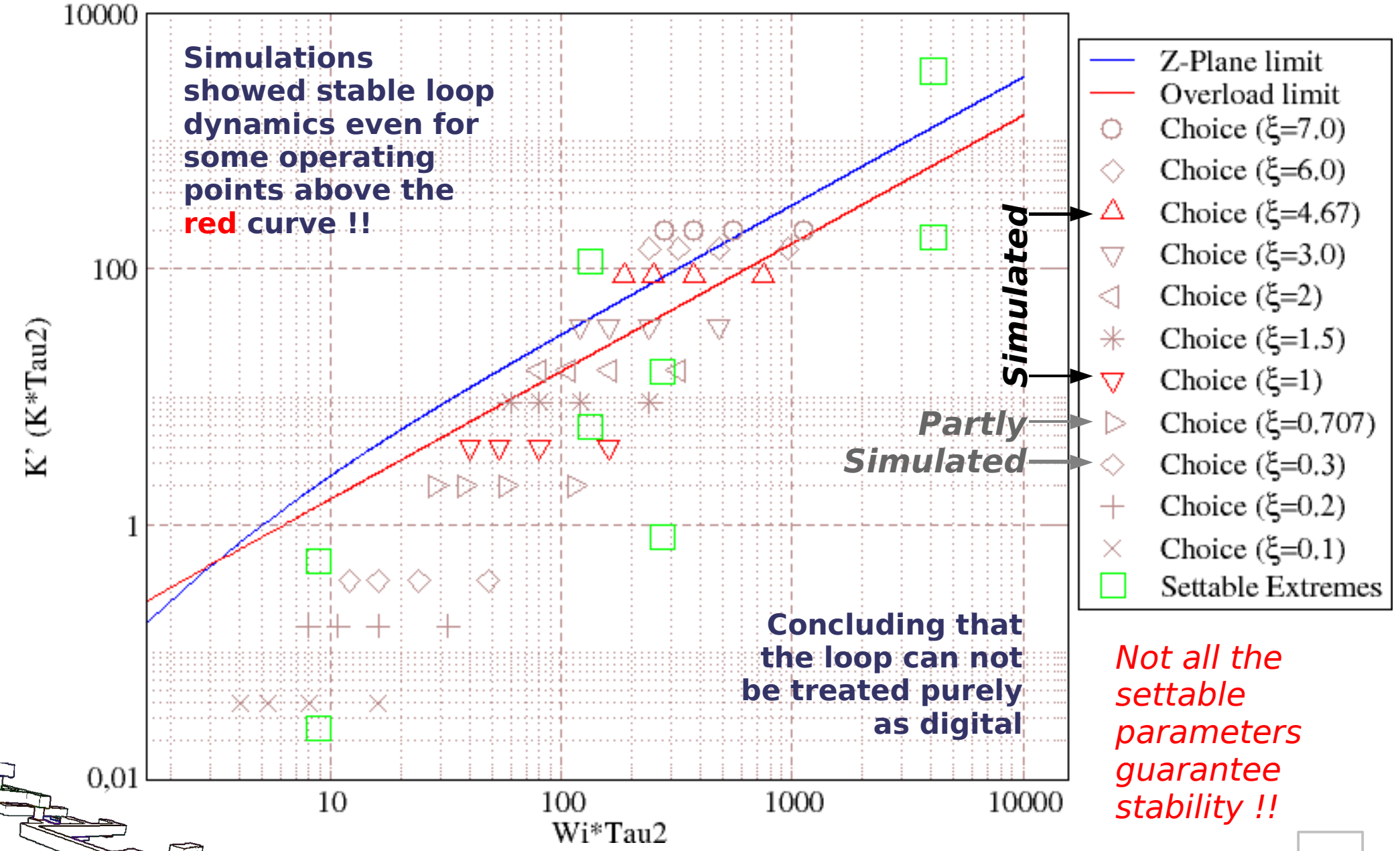
- Run the above m-file for 16 parameter sets to produce the ***root loci***, and ***bode*** plots, ***impulse*** and ***step*** responses of the ***transfer functions***



(high resolution images, zoom in)

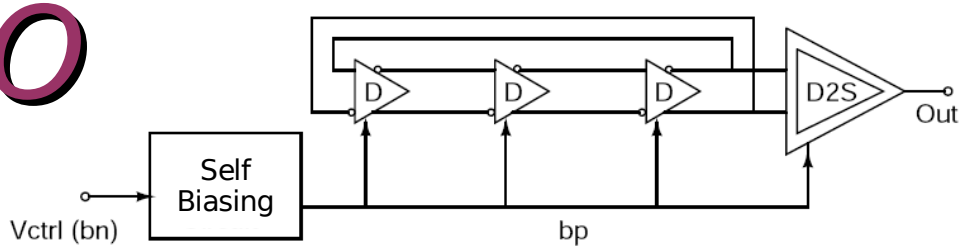
CP-PLL Stability Limits

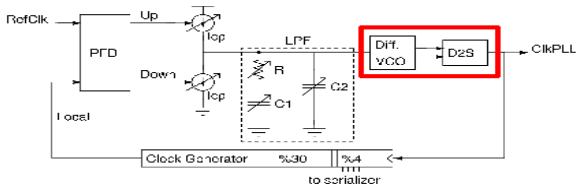
Theory & Practice

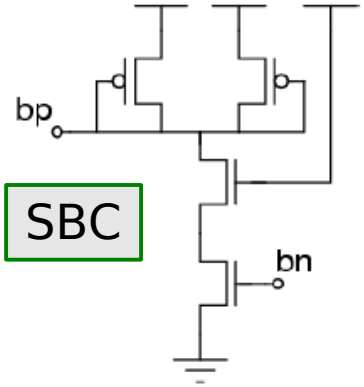


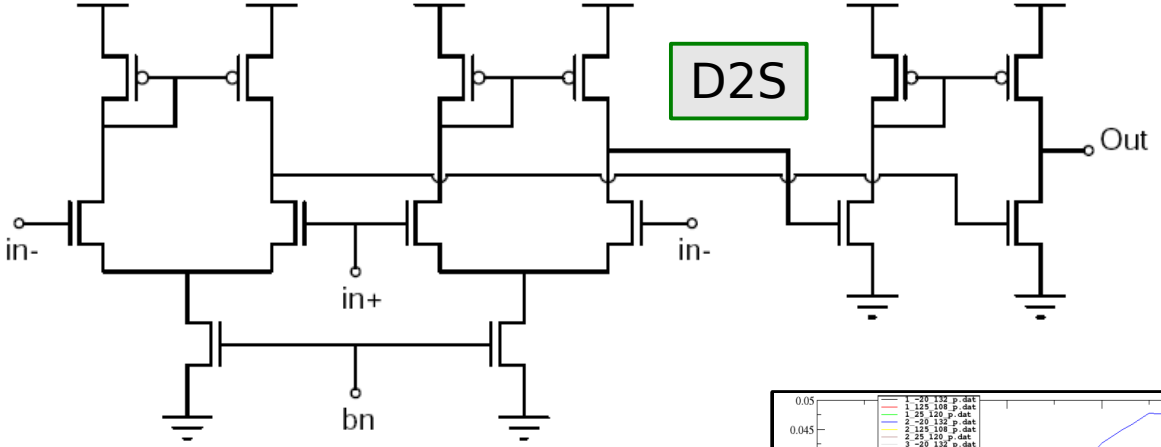
Not all the settable parameters guarantee stability !!

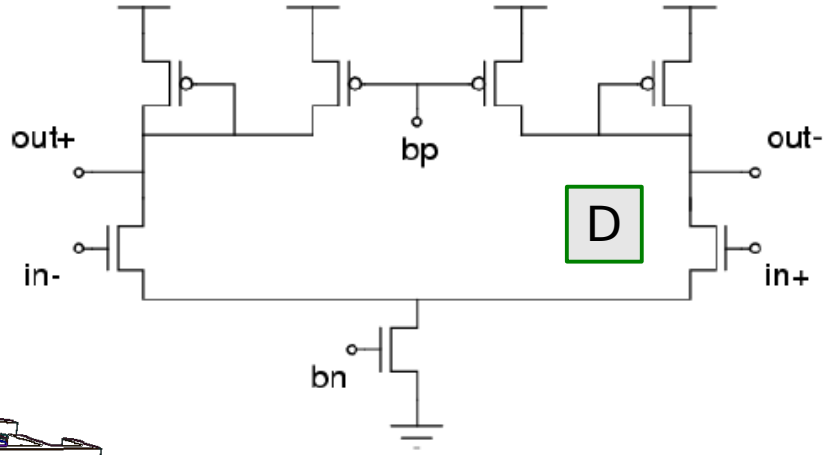
VCO

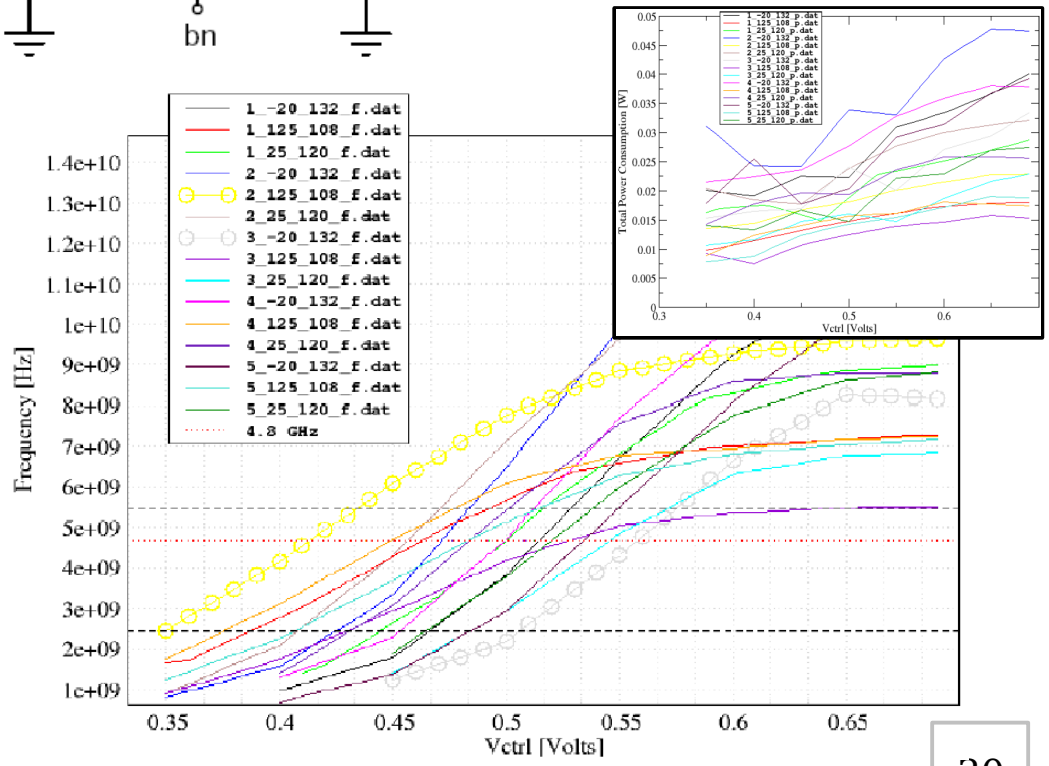

D
D
D
D2S
Out
Self
Biasing
Vctrl (bn)
bp

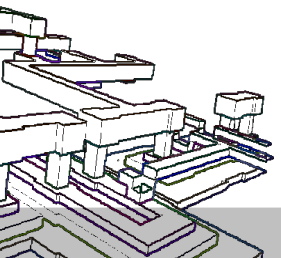

RefClk
Up
Icp
PFD
Down
Icp
LPF
R
C1
C2
Diff.
VCO
D2S
ClkPLL
Local
Clock Generator
%30
%4
to serializer


bp
SBC
bn

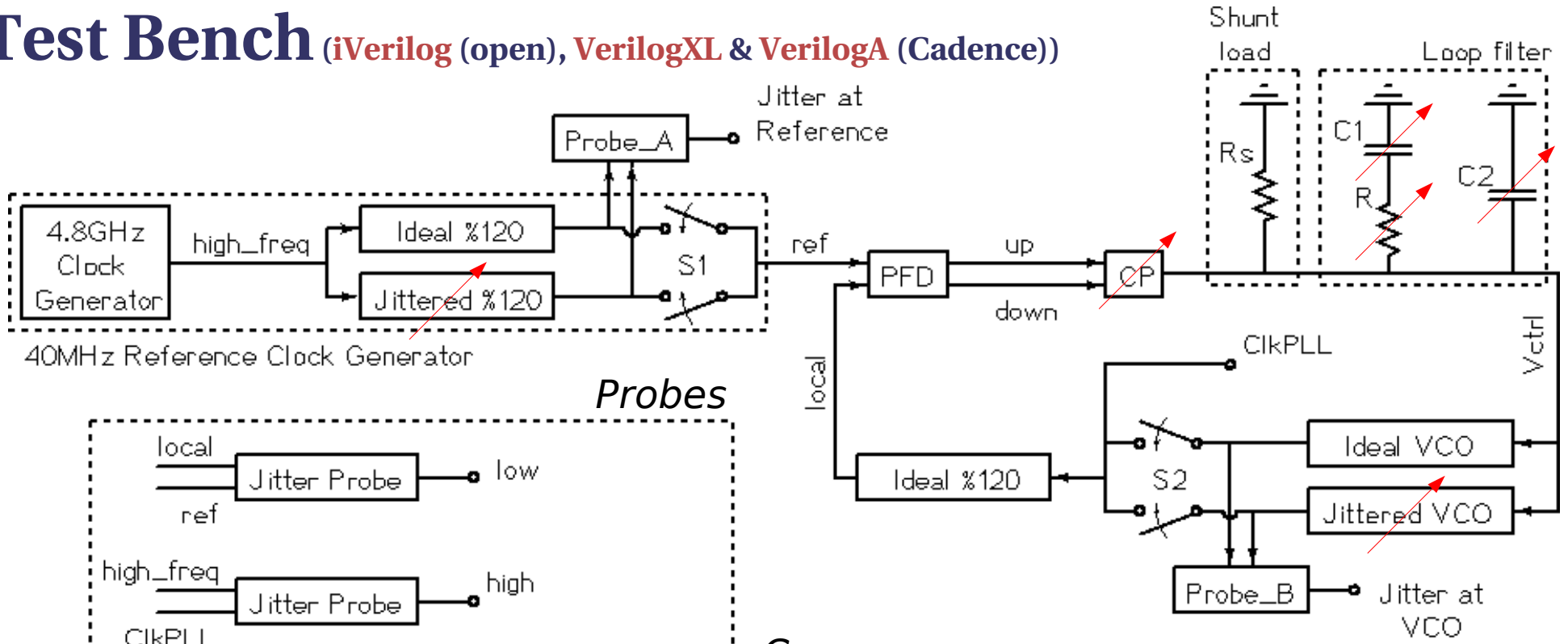

D2S
Out
in-
in+
in-
bn


out+
bp
out-
D
in-
in+
bn


1_-20_132_f.dat
1_125_108_f.dat
1_25_120_f.dat
2_-20_132_f.dat
2_125_108_f.dat
2_25_120_f.dat
3_-20_132_f.dat
3_125_108_f.dat
3_25_120_f.dat
4_-20_132_f.dat
4_125_108_f.dat
4_25_120_f.dat
5_-20_132_f.dat
5_125_108_f.dat
5_25_120_f.dat
4.8 GHz
Frequency [Hz]
Vctrl [Volts]
Total Power Consumption [W]



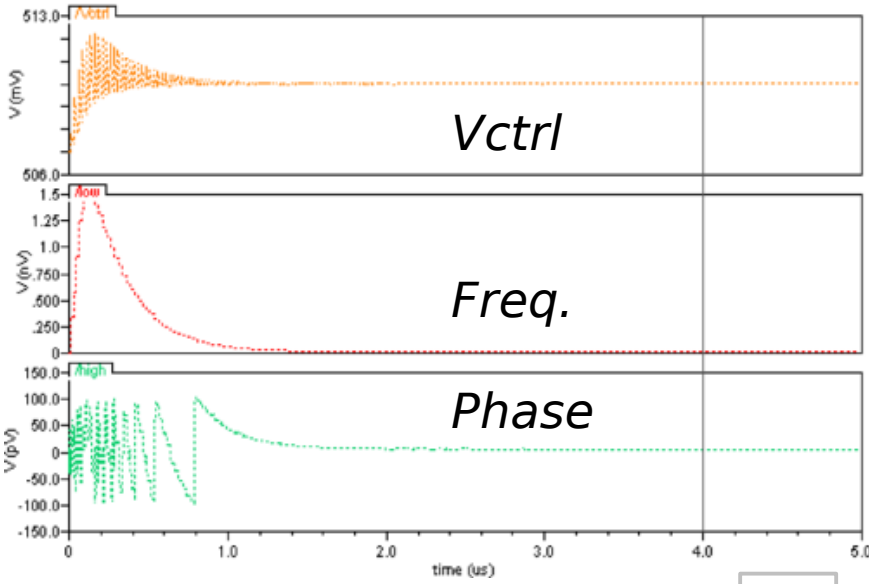
Test Bench (iVerilog (open), VerilogXL & VerilogA (Cadence))

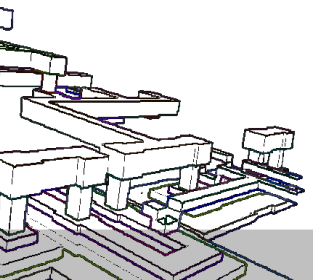


Corners

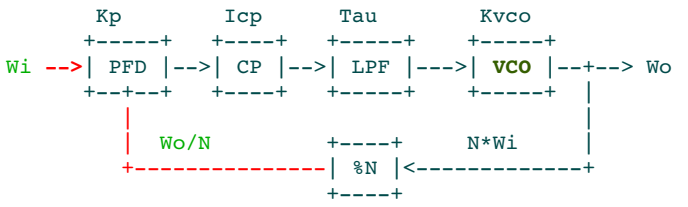
Table 5.3: Technology process corners used for parameter extraction.

Index	Corner	T [C^o]	Vdd [V]	Abbreviation
0	1	125	1.08	1_125_108
1	1	25	1.2	1_25_120
2	1	-20	1.32	1_-20_132
3	2	125	1.08	2_125_108
4	2	25	1.2	2_25_120
5	2	-20	1.32	2_-20_132
6	3	125	1.08	3_125_108
7	3	25	1.2	3_25_120
8	3	-20	1.32	3_-20_132
9	4	125	1.08	4_125_108
10	4	25	1.2	4_25_120
11	4	-20	1.32	4_-20_132
12	5	125	1.08	5_125_108
13	5	25	1.2	5_25_120
14	5	-20	1.32	5_-20_132

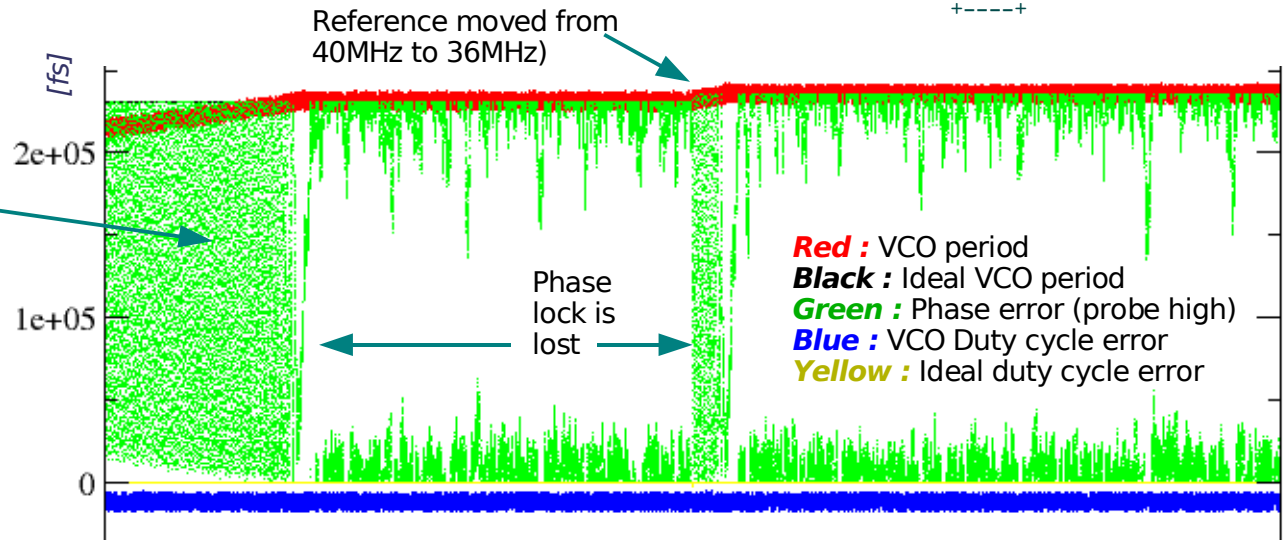




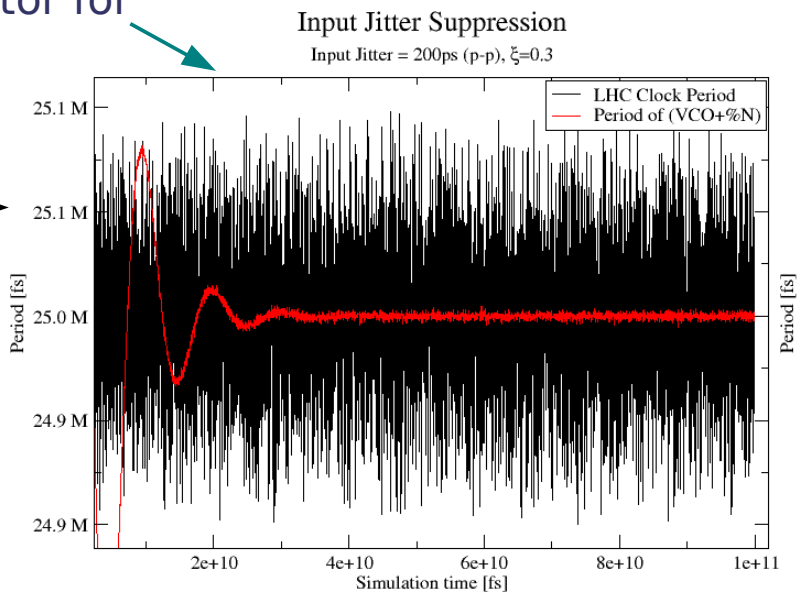
Simulations (iVerilog (open), VerilogXL & VerilogA (Cadence))

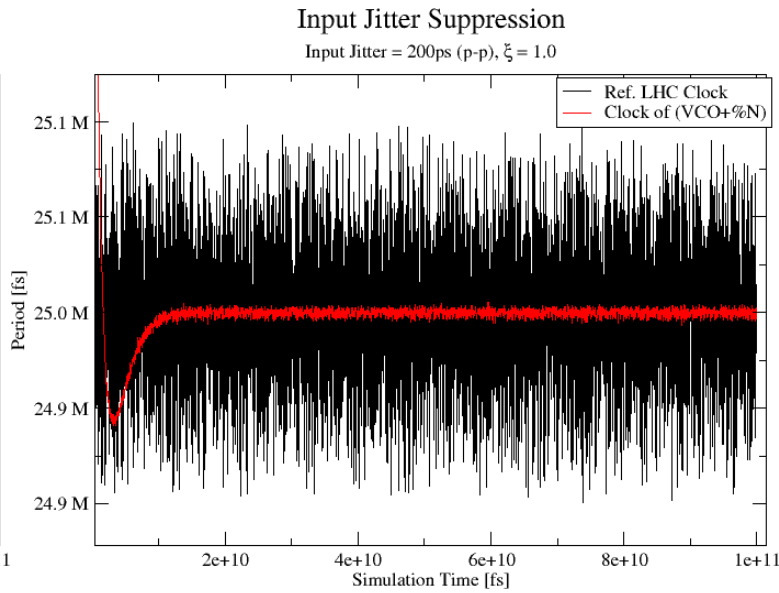


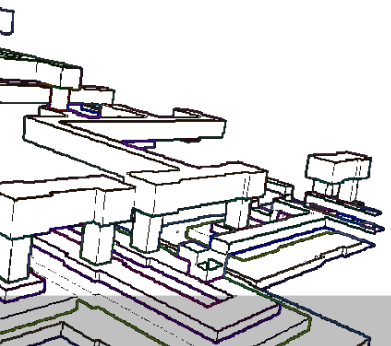
- Very noisy VCO + very noisy reference
- Initially not locked
- Reference frequency step of 10% some time after locking
- Low bandwidth CP-PLL filters out the noise at the reference input (i.e. slow loop)
- Effect of damping factor for 0.3 and 1.0



Red : Period of VCO+%N
Black : Reference period with 200ps (p-p) jitter







(high resolution images, zoom in)

Cores #2 (Octave m-files)

- Simulation results (one dimensional data files or vectors of instant period for locked state) are evaluated with the following m-files (high resolution images, zoom in) :

periodMeter.m - /home/oc/Documents/PhD/HDL/VerilogA/Verilo...

```
% produces plots for the output of the PLL

clear;

load out/periodMeter.dat;
offset = 100000;

T = Center = mean(periodMeter);
Sigma  = std(periodMeter);
maxdT = max(abs(periodMeter-T))/T;

fprintf("Center = %.9g, 1/Center = %.9g\n", T, 1/T);
fprintf("Sigma_abs = %.9g, Sigma_rel = %.9g%%\n", Sigma, 100*Sigma/T);
fprintf("Max dT = %.9g%%\n", 100*maxdT);
fprintf("Absolute Jiier (aj) = %.9g\n", aj(periodMeter, Center, offset));
fprintf("Cycle-to-cycle Jiier (cc) = %.9g\n", cc(periodMeter, offset));


plot (periodMeter); figure;
hist (periodMeter, 200); figure;

pause;
```

cc.m - /home/oc/Documents/PhD/HDL/VerilogA/Veril...

```
% returns the cycle-to-cycle jitter of the vector

function retval = cc(vector, offset)
    retval = 0;
    if (nargin != 2)
      usage ("cc (vector, offset)");
    endif
    if (isvector (vector))
      length = length(vector);
      i = offset;
      sum = 0;
      while(i<length)
        difference = vector(i+1) - vector(i);
        sum = sum + difference * difference;
        i = i + 1;
      endwhile
      retval = sqrt(sum)/length;
    else
      error ("ERROR : cc : Expecting a vector argument !..");
    endif
endfunction
```

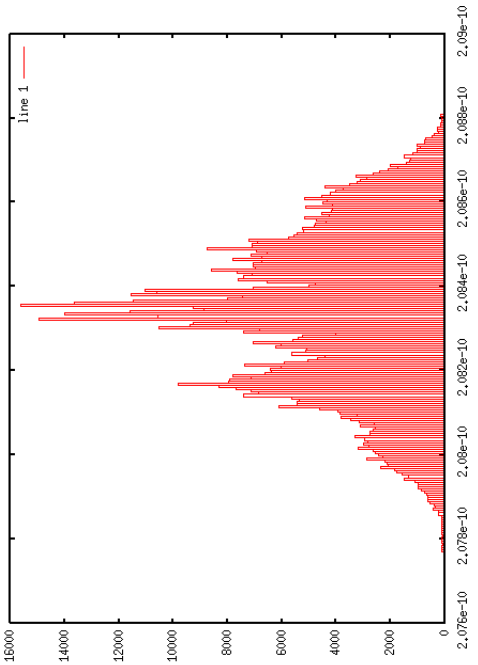
aj.m - /home/oc/Documents/PhD/HDL/VerilogA/Veril...

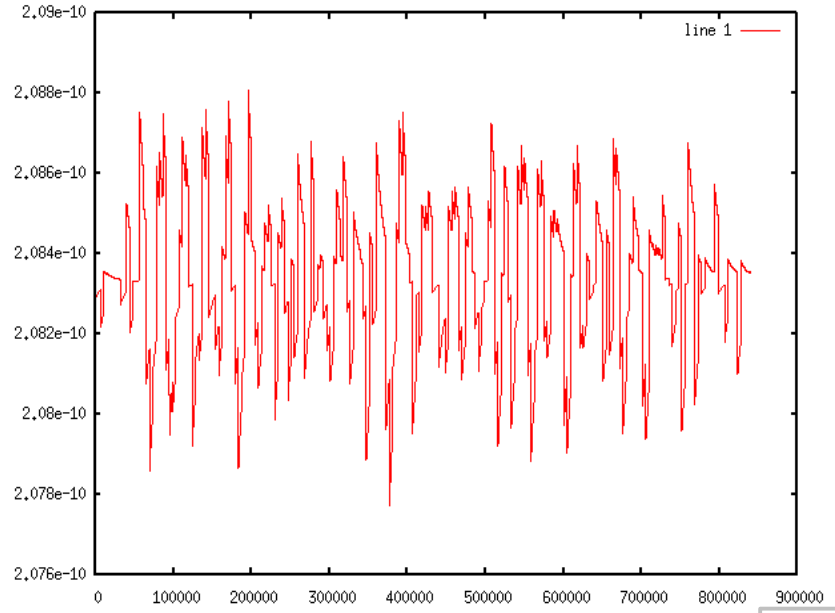
```
% returns the absolute jitter of the vector

function retval = aj(vector, average, offset)
    retval = 0;
    if (nargin != 3)
      usage ("aj (vector, average, offset)");
    endif
    if (isvector (vector))
      length = length(vector);
      i = offset;
      sum = 0;
      while(i<length)
        product = vector(i) - average;
        sum = sum + product * product;
        i = i + 1;
      endwhile
      retval = sqrt(sum)/length;
    else
      error ("ERROR : aj : Expecting a vector argument !..");
    endif
endfunction
```

- These are the m-files to calculate the ***statistics***, plot the the ***periods*** of interest as a function of simulation time (also in ***histogram*** form) as seen below :

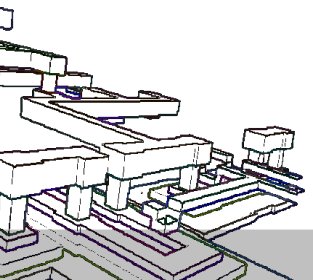
Statistics :

Center = 2.08333538e-10
1/Center = 4.79999528e+09
Sigma_abs = 1.87044613e-13
Sigma_rel = 0.0897813257%
Max dT = 0.27241816%
Absolute Jitter = 1.91269e-16
Cycle-to-cycle Jitter = 2.70e-18



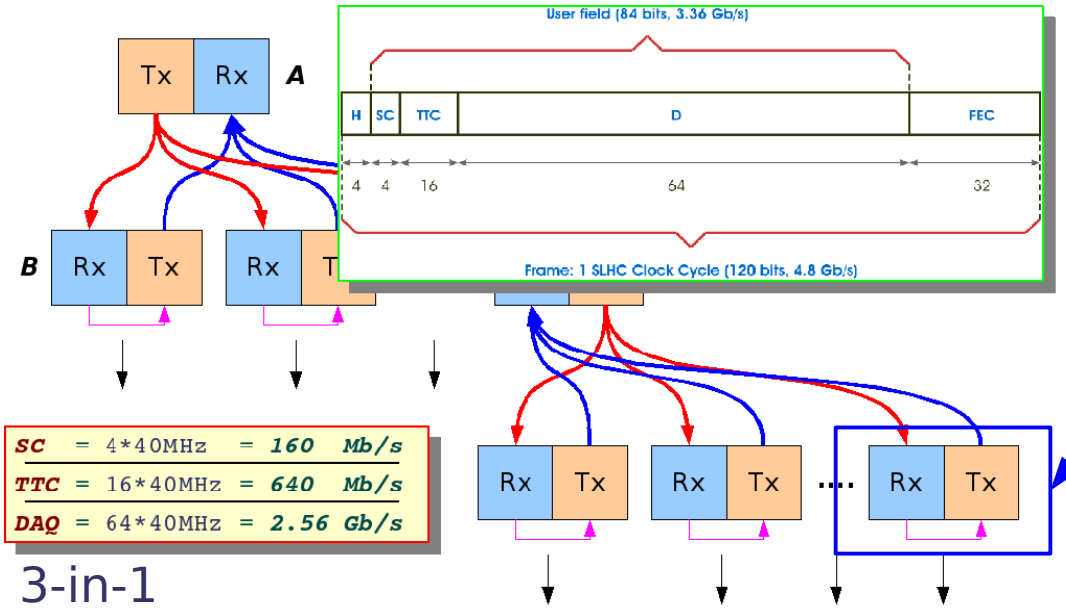


(high resolution images, zoom in)



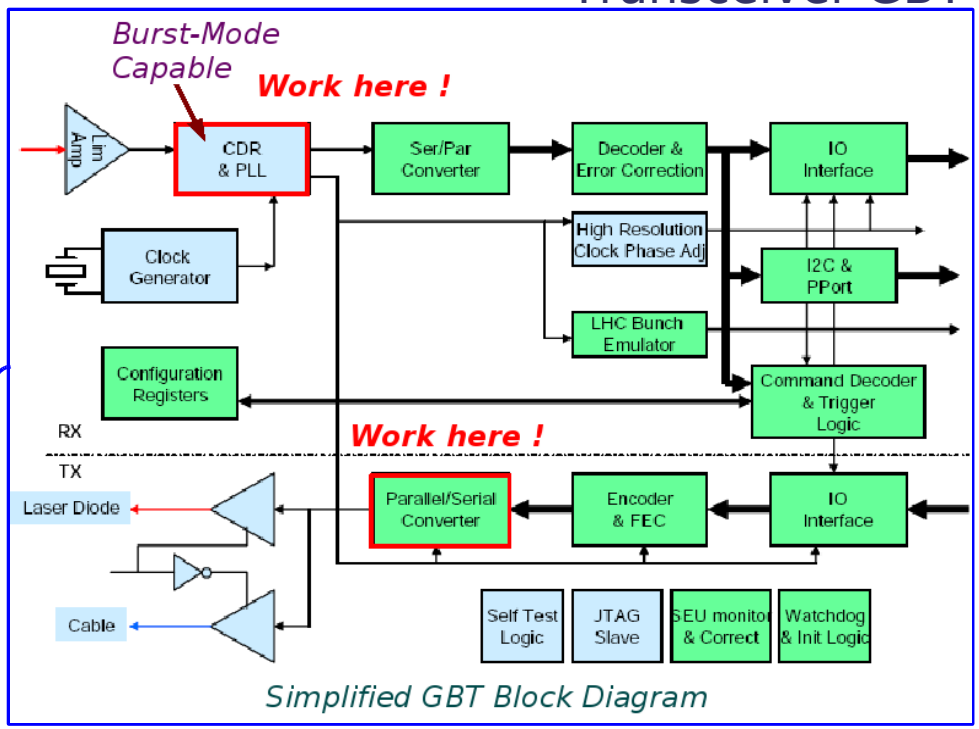
GBT-13

Line Format

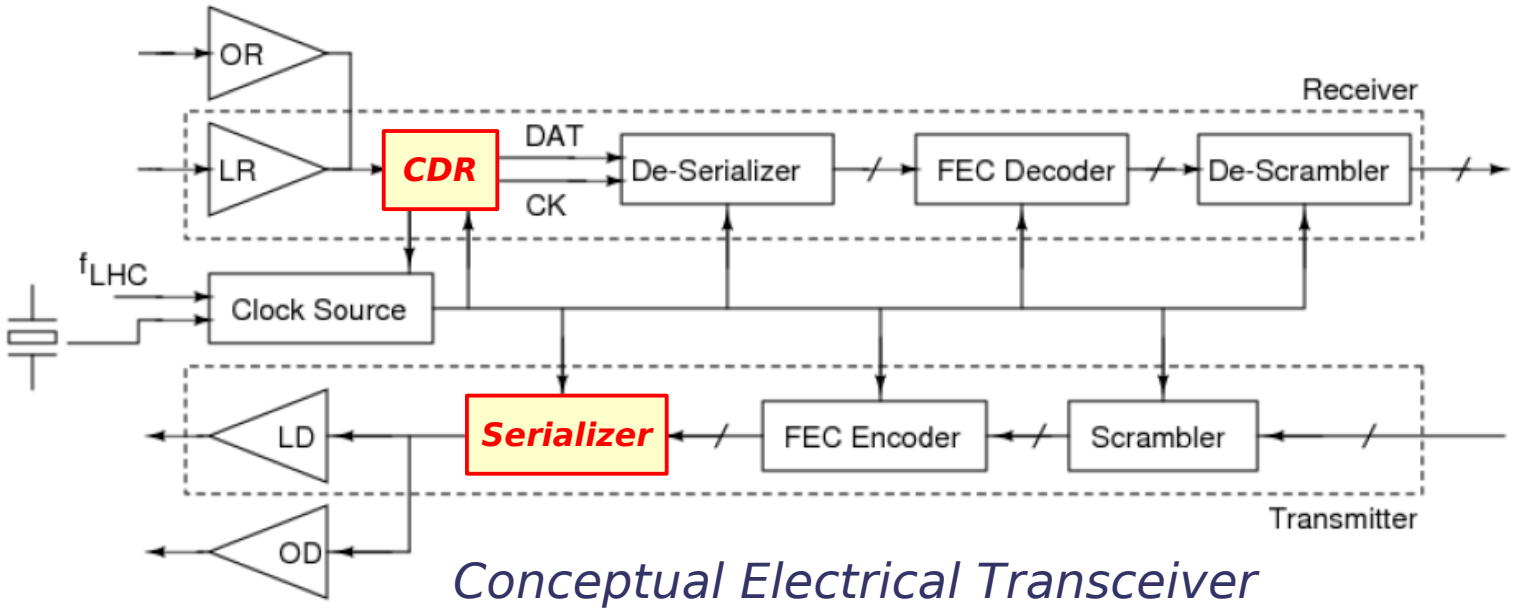


3-in-1

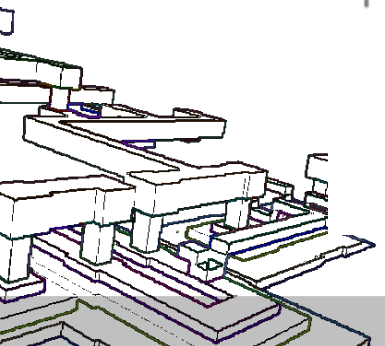
Transceiver GBT



Simplified GBT Block Diagram



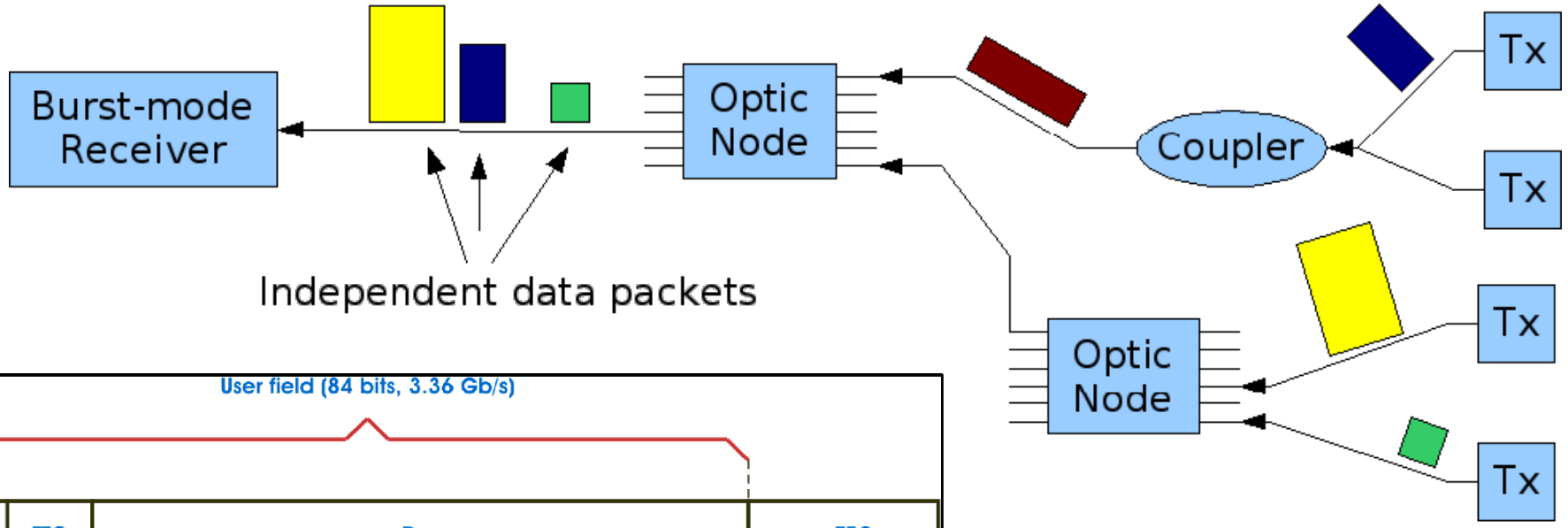
Conceptual Electrical Transceiver

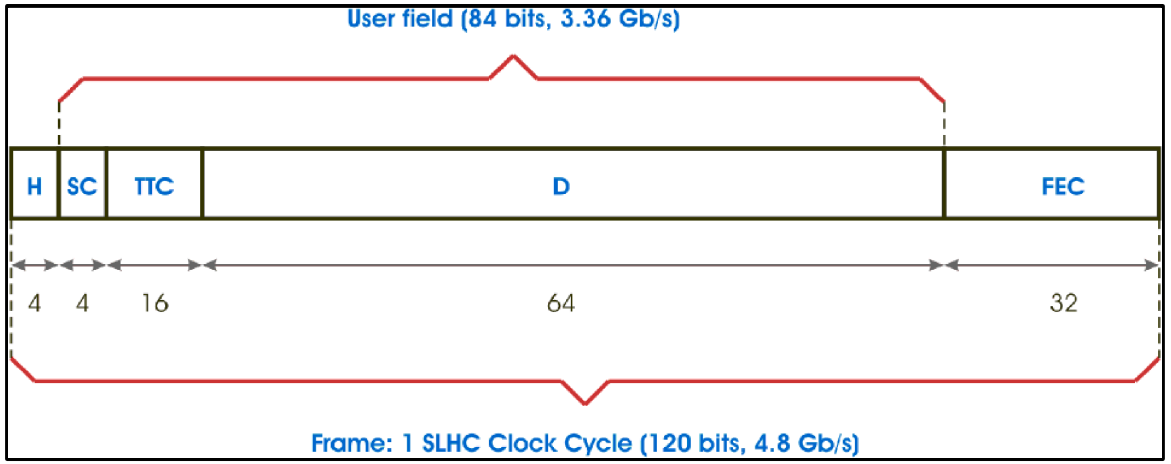


Burst-CDR for GBT of S-LHC

- **Development Target**
  - *Burst-Mode capable Gb/s class Clock and Data Recovery (CDR) chip of Giga Bit Transceiver (GBT) for S-LHC @ CERN*
  - ***My contributions** is modeling, designing, implementing, and testing of burst-mode capability building block of the full CDR*

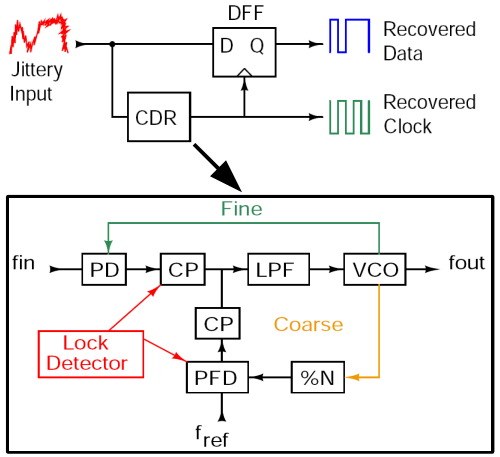
- *Packet based switching & routing*
- *Packets with different power & time delay, fluctuate dynamically*
- *Each package could have a header (preamble) to be used by the receiver for synchronization*
- *Payload is encoded (e.g. Manchester or Reed-Solomon) requiring a decoder*



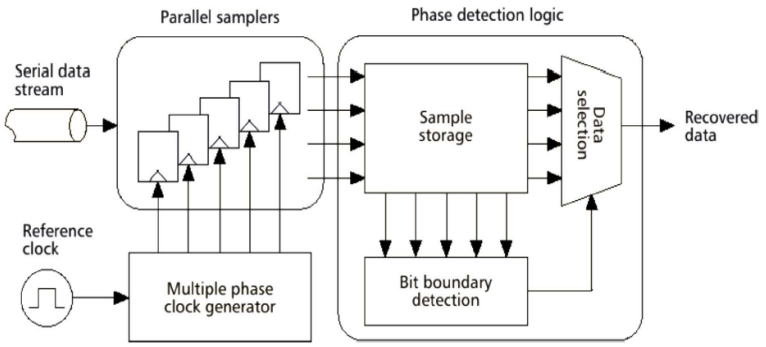


CDR Classification

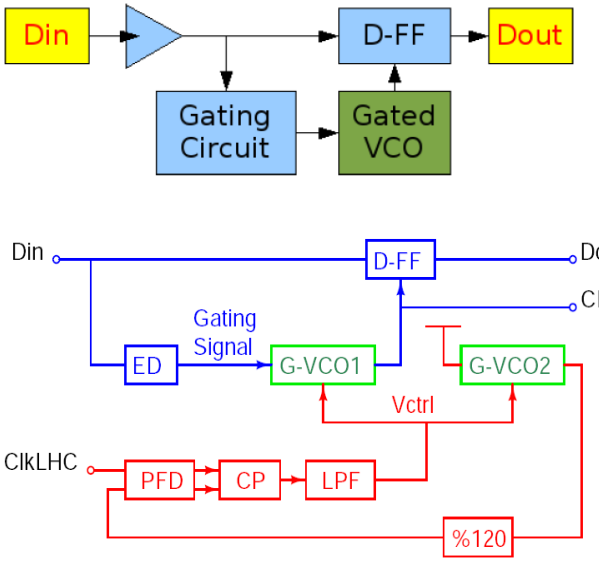
1) PLL-Based



2) Blind Oversampling



3) Gated VCO

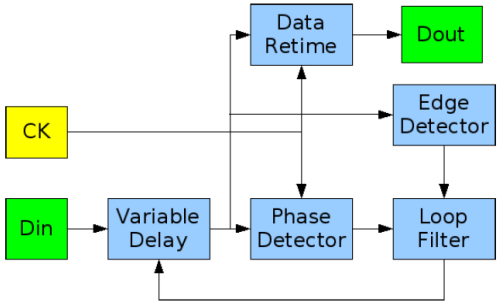


Behaviorally CDRs are:

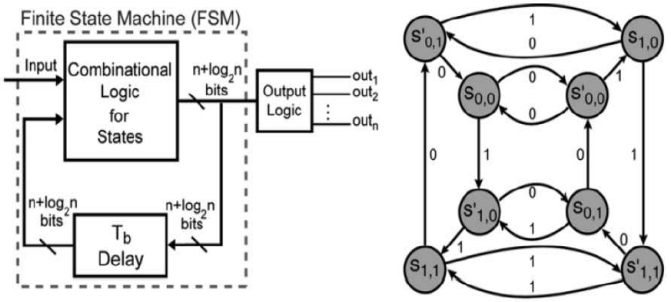
- *Continuous* **vs** *Burst*
- *Closed* **vs** *Open Loop*
- *Filter* **vs** *Over-Sampling*
- *Delay Clock* **vs** *Data*
- *Digital* **vs** *Analog*

4) DLL-Based

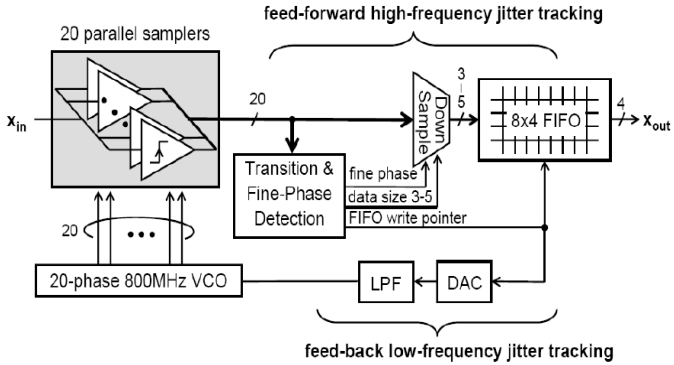
(b)Data-De-skew

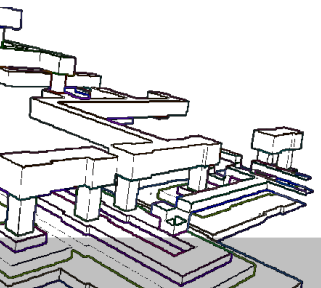


5) FSM Based



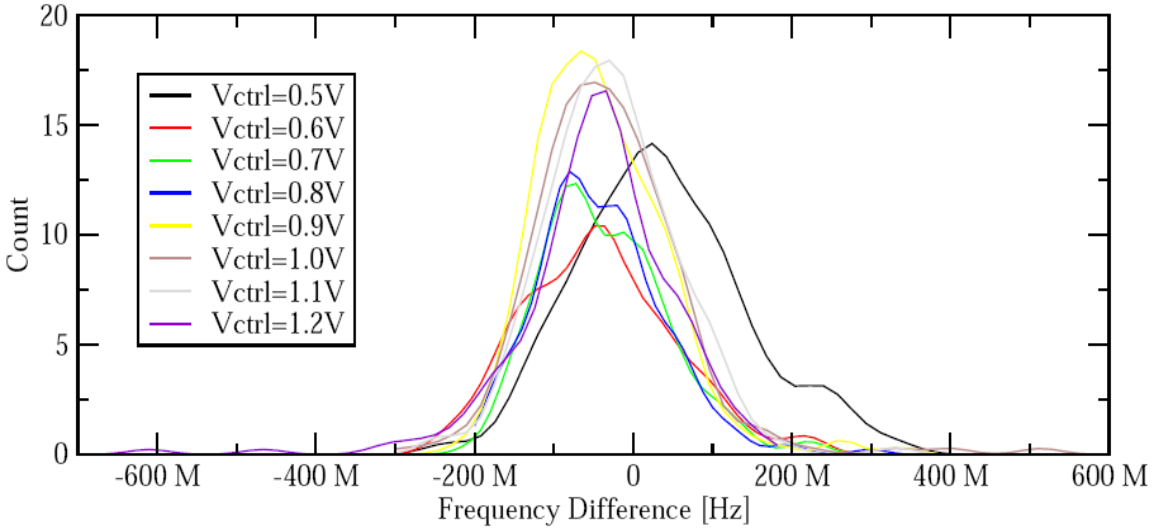
6) Hybrid

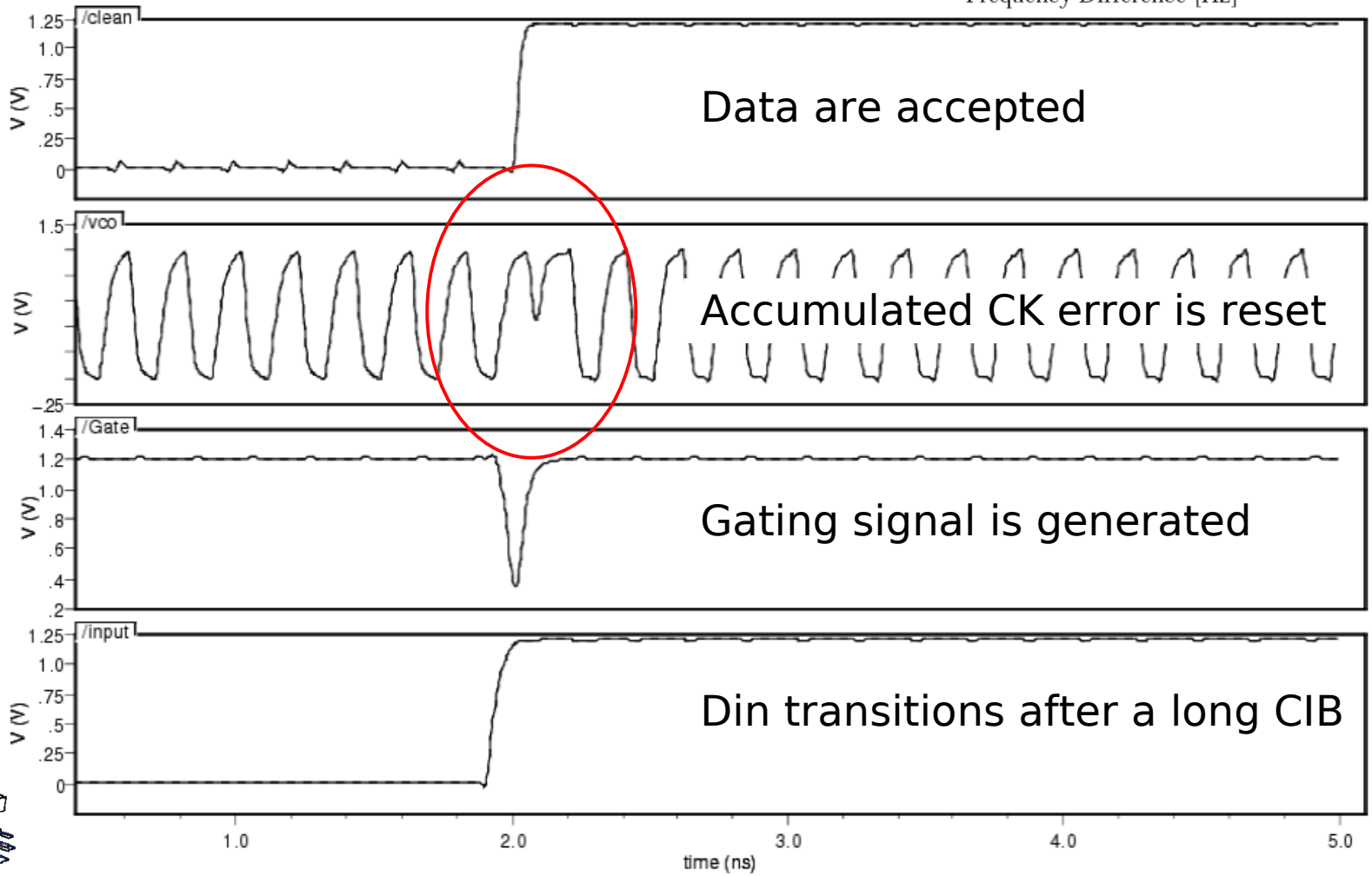


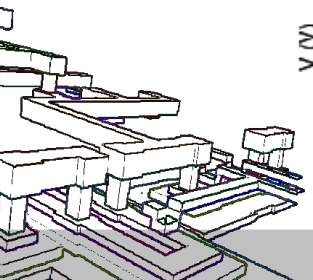


Simulation

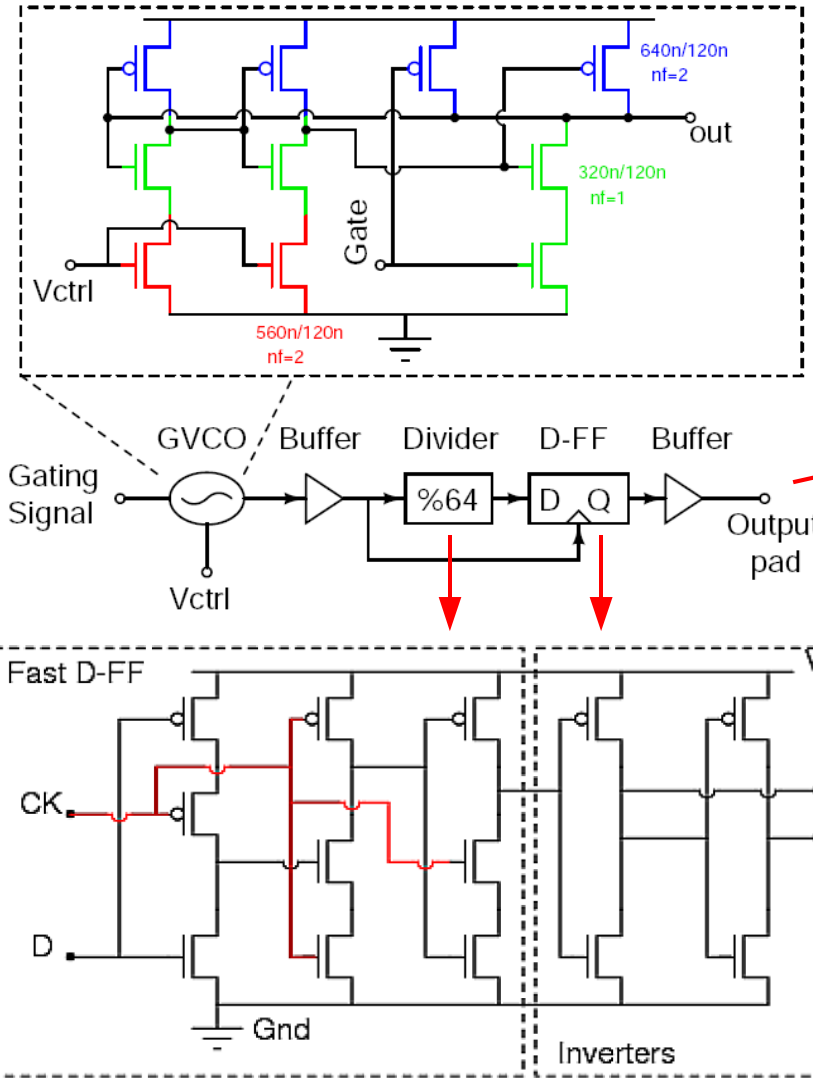
Condition is met:
10% MC worst-case frequency difference, thus a tolerance of 10 CIB in a random input data stream

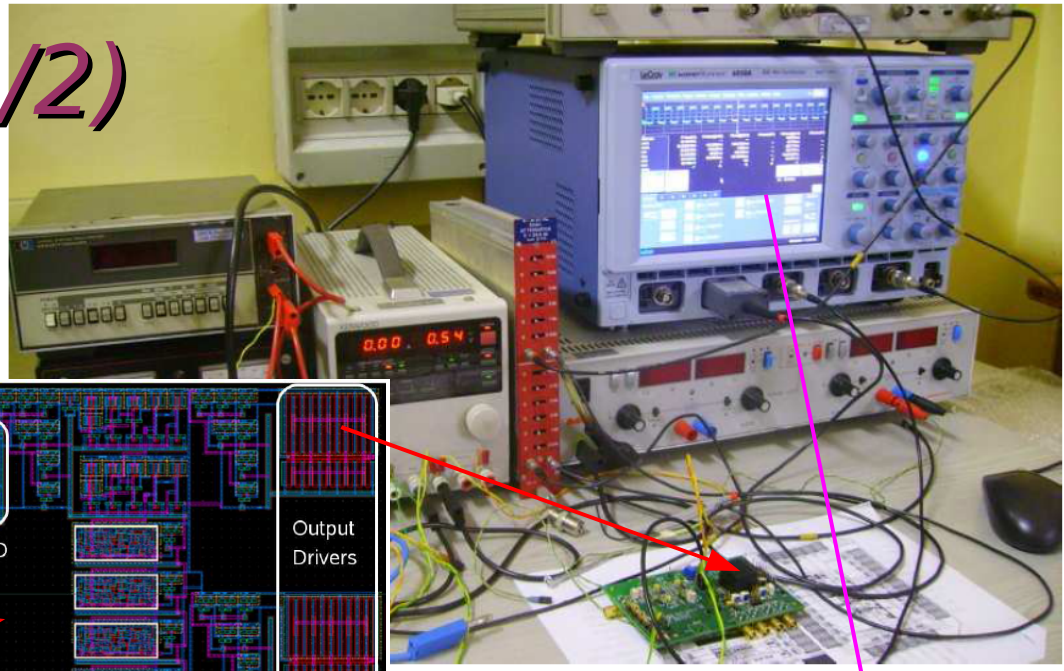


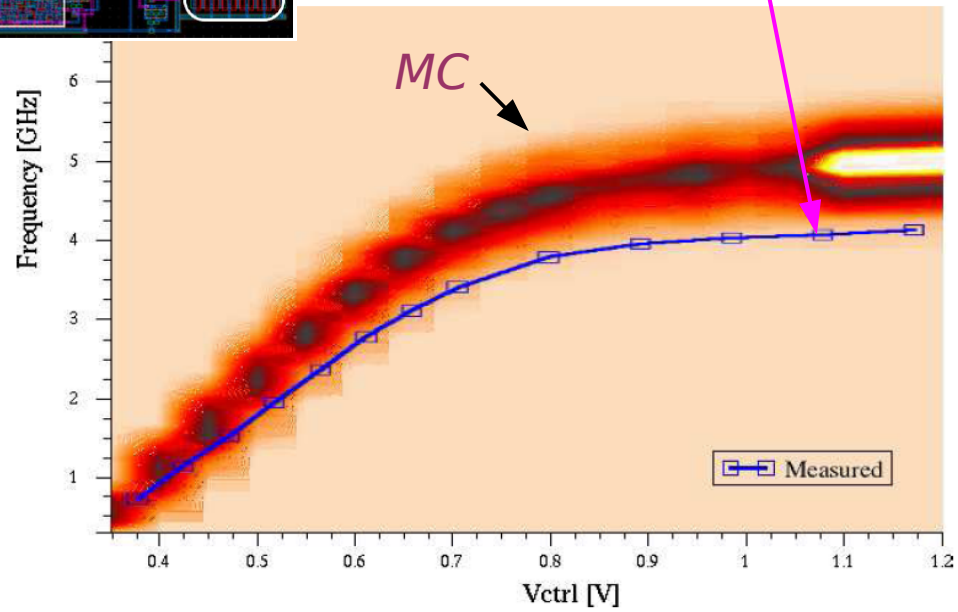


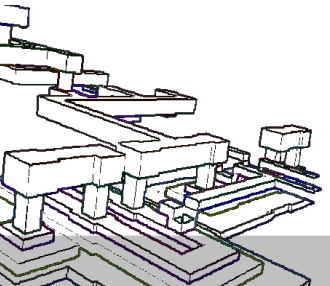


Measurement (1/2)



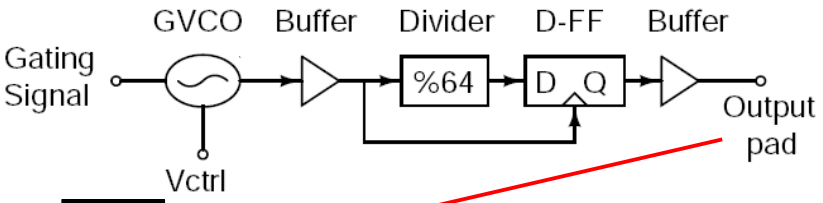


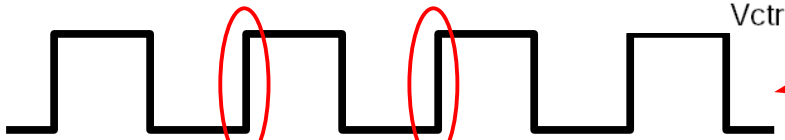




- *Effect of pre-fabrication steps, such as **chip filling***
- *Accuracy of the models: the **cross-section** of low and high frequency models of the transistors*
- *Issues relating to the measurement system itself*

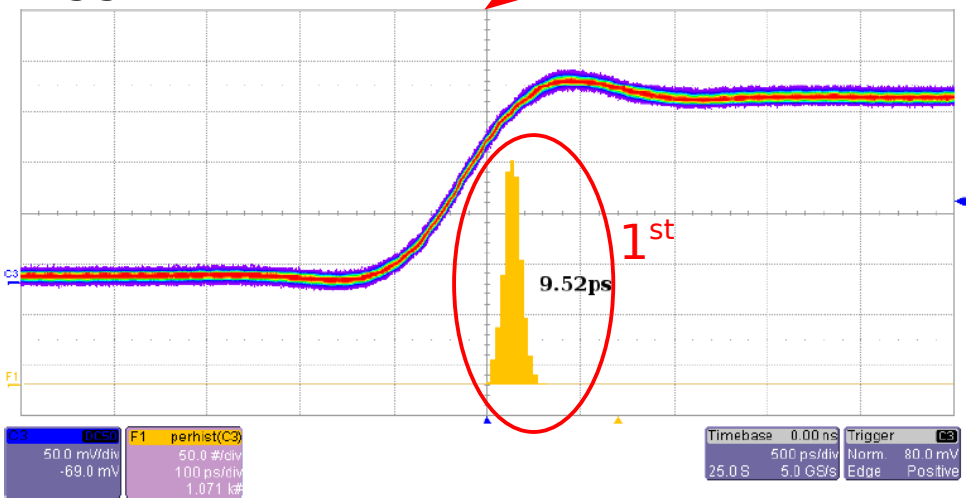
Measurement (2/2)

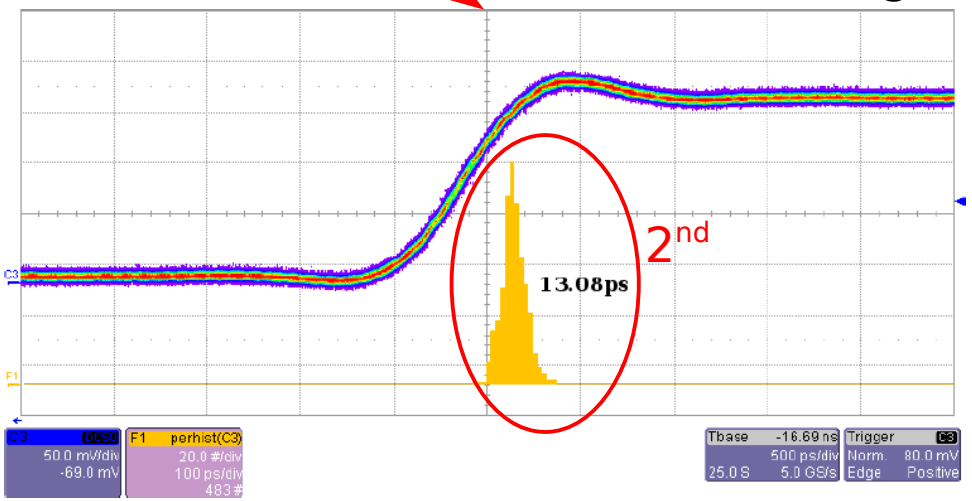




Oscilloscope triggers

2nd rising edge

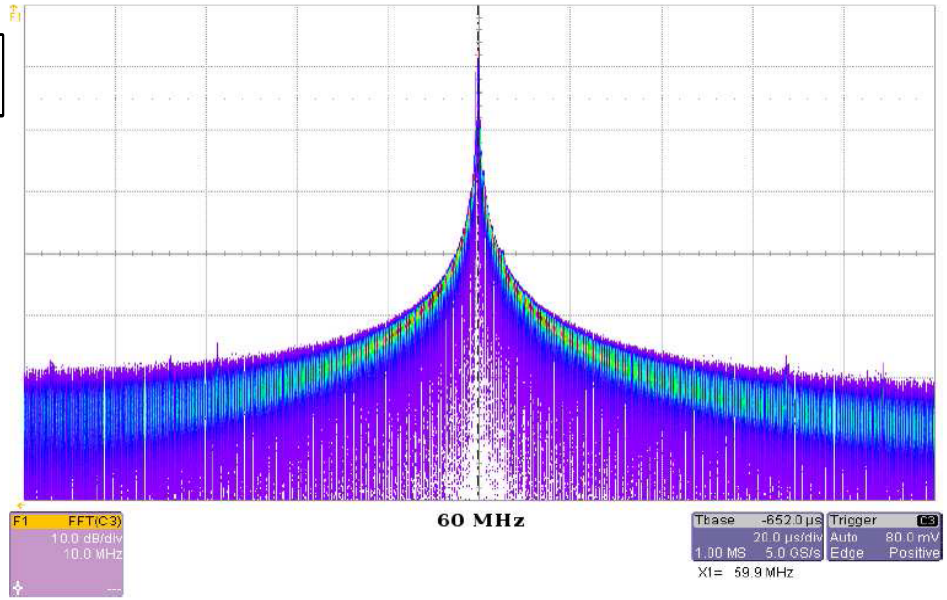


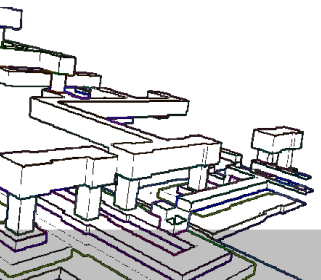


$$J_{VCO,C2C} = \sqrt{13.08^2 - 9.52^2} = 8.97\,ps$$

2nd 1st

Standard deviation of cycle-to-cycle jitter associated with the VCO



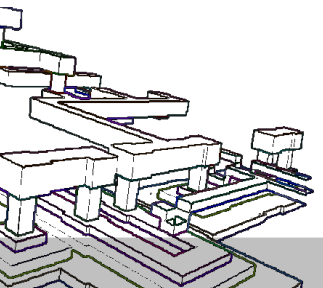


Conclusion (2/2)

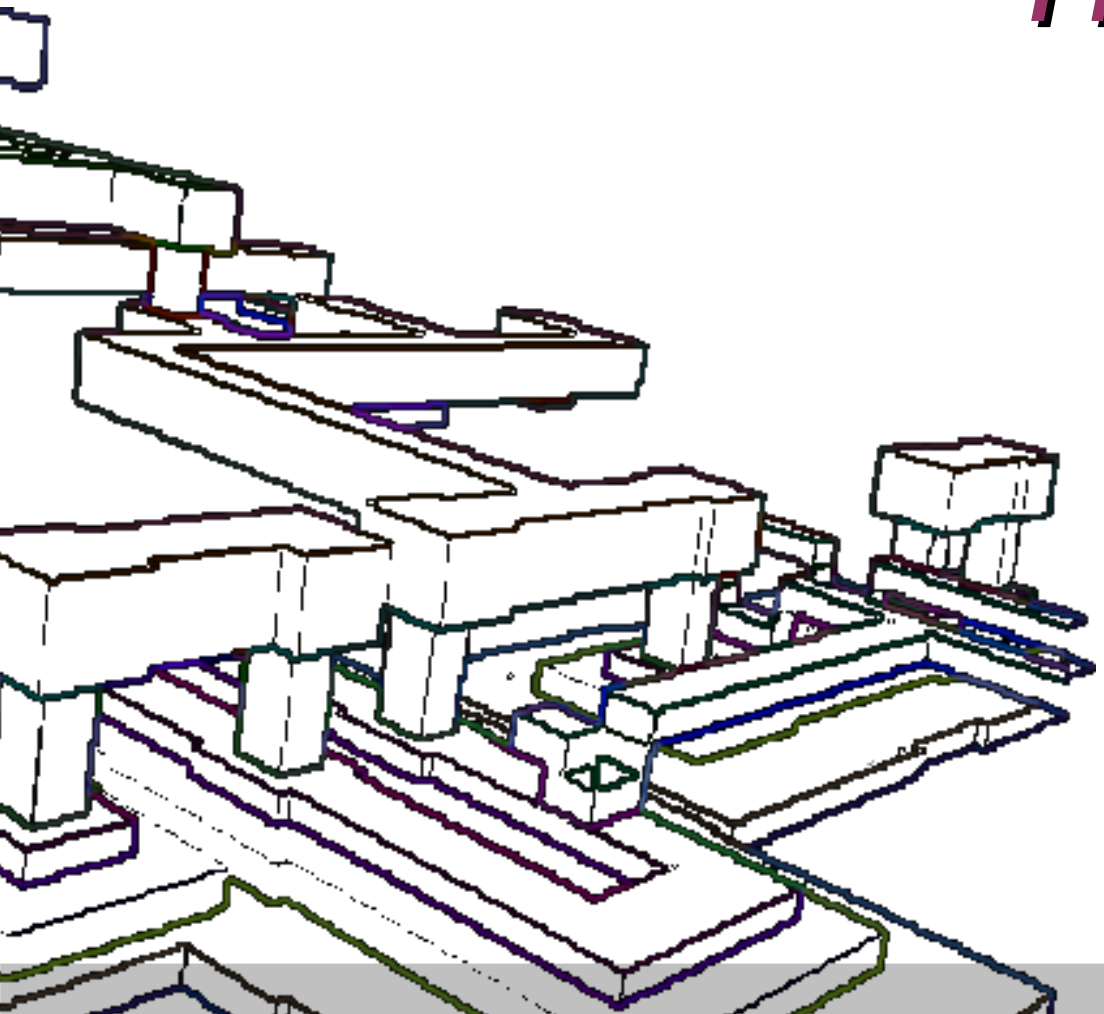
- The Project
  - Early stages
  - Needs Multiple Prototypes
  - Currently Complete Sub-Blocks or Test Structures on Shared Dies

- CP-PLL based Serializer
  - Full Transistor Level Implementation Done
  - 130nm, CMOS
  - CP-PLL Loop ***Parametrization*** and Evaluation with CaPPeLLo

- Burst-Mode CDR
  - Currently no decision on architecture by the collaboration
  - I used two adopted architectures: CP-PLL Based Gated-VCO
  - Currently not full speed, needs engineering



Thank you.



Introduction (1/2)

HEP path is :

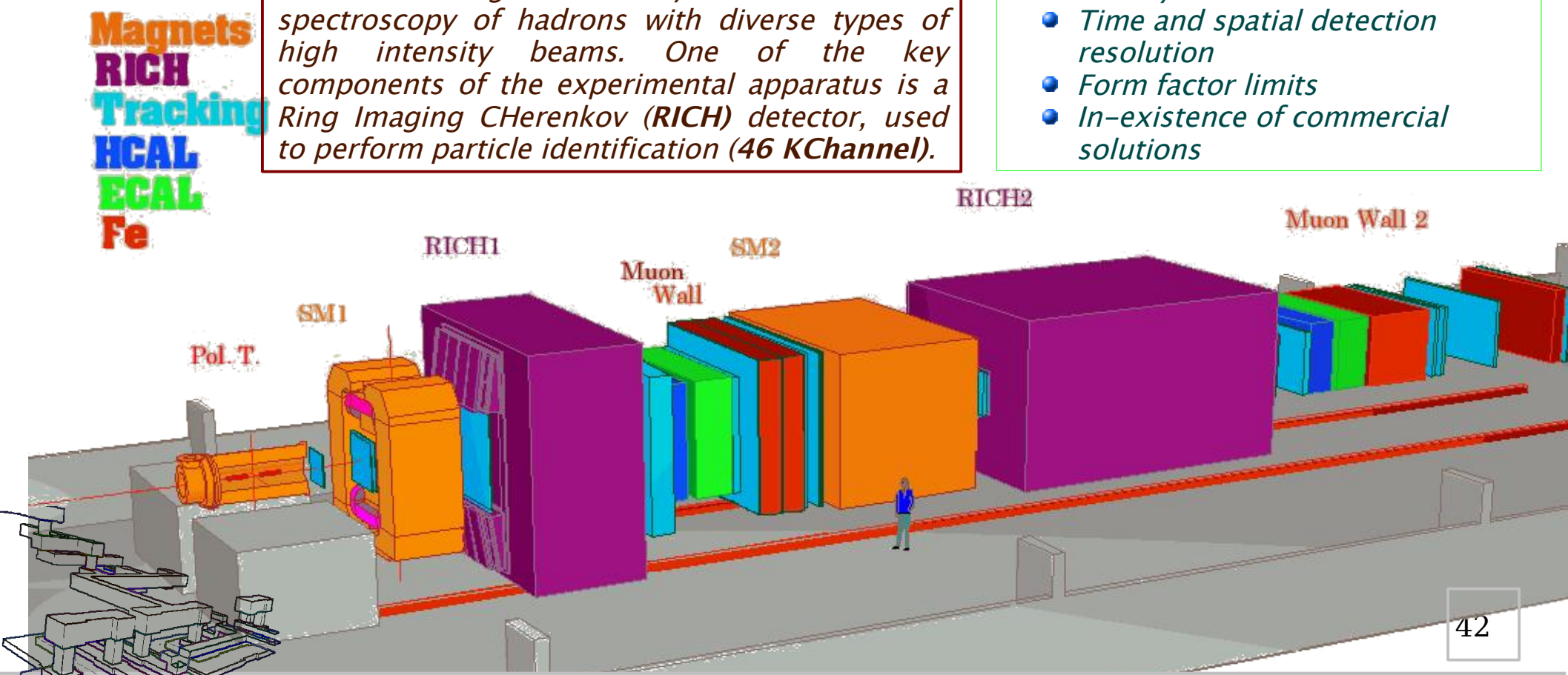
- *...to accelerate particles through particle accelerators,*
- *...to collide them at high energies within large detector systems,*
- *...to detect the interactions of the particles in the form of "**events**",*
- *...to store **huge** amount of data (Peta Byte per year),*
- *...to process this data and analyze **physics events**,*
- *...to understand the **physics** of sub-atomic world.*

Example :

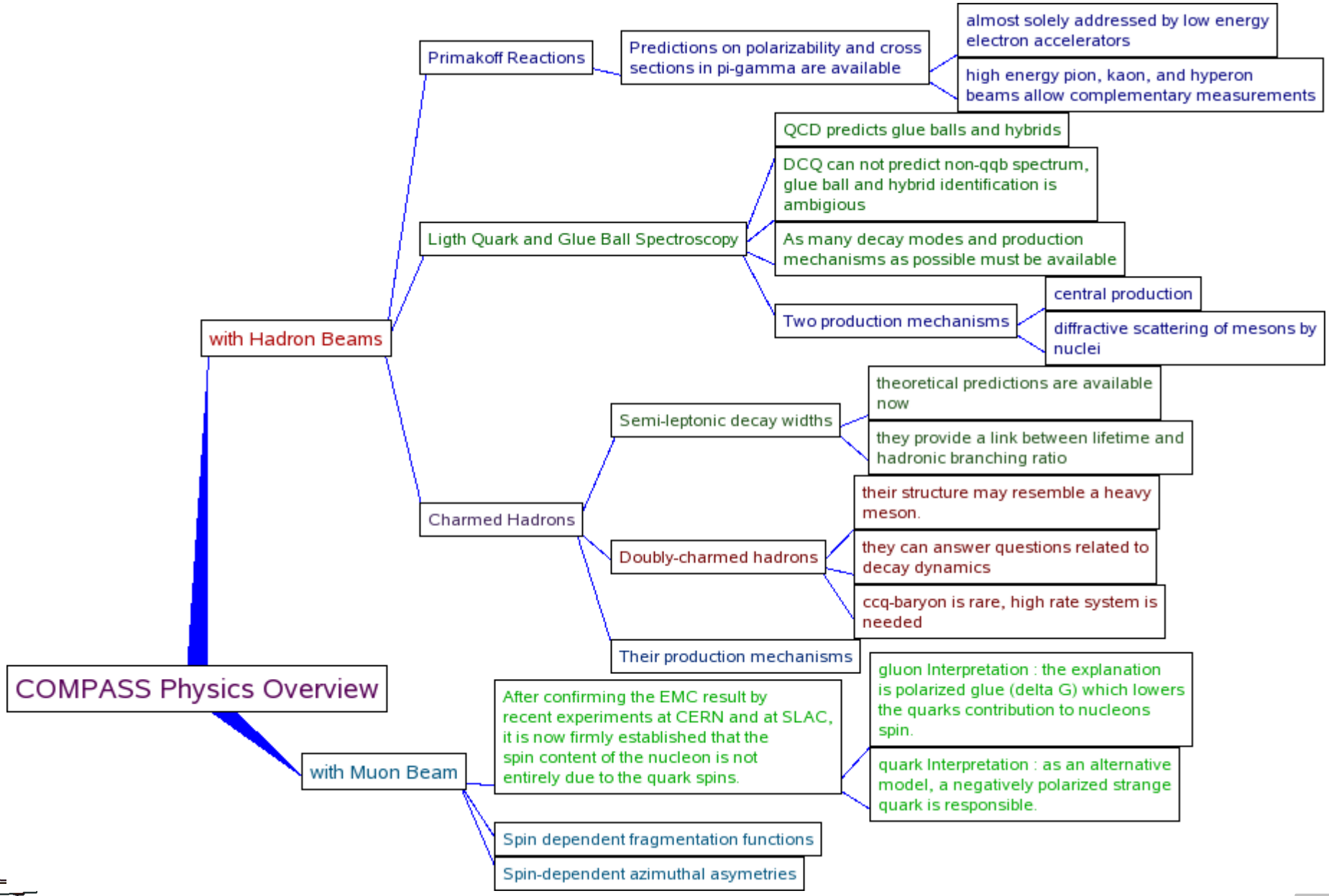
***COMPASS** is an experiment at the **CERN SPS** designed to study the structure and spectroscopy of hadrons with diverse types of high intensity beams. One of the key components of the experimental apparatus is a Ring Imaging CHerenkov (**RICH**) detector, used to perform particle identification (**46 KChannel**).*

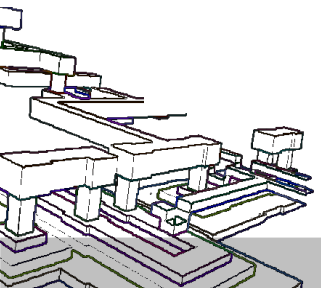
Why full custom ASIC ?

- *Heavily radioactive environment*
- *Time and spatial detection resolution*
- *Form factor limits*
- *In-existence of commercial solutions*



Back-up Pages

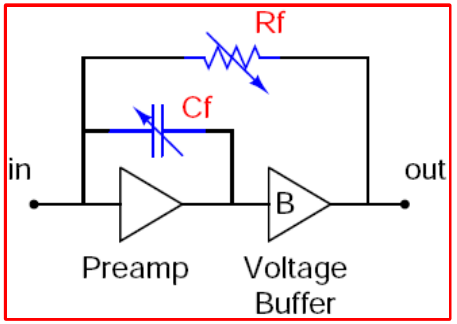


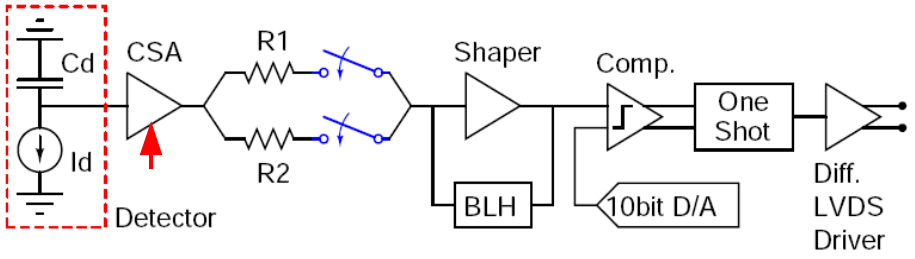


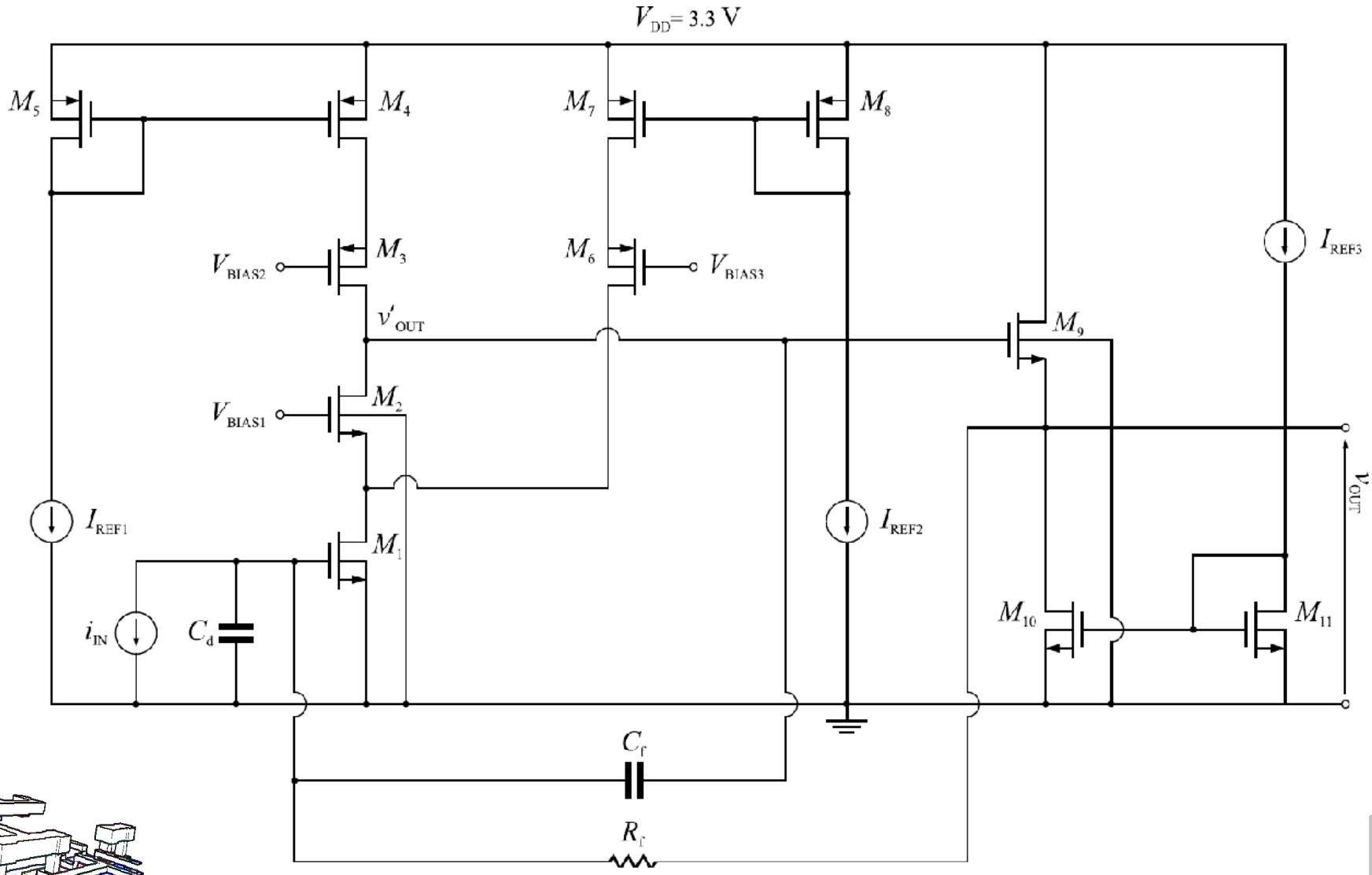
Blocks

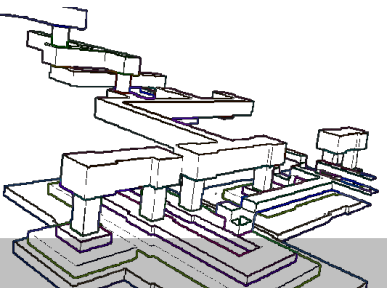
CSA

1st Stage





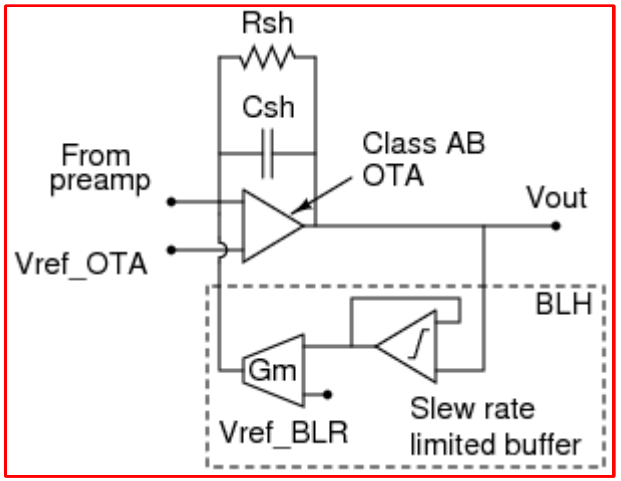


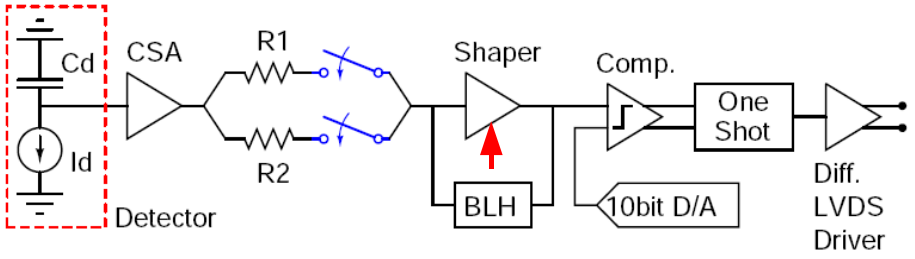


Blocks

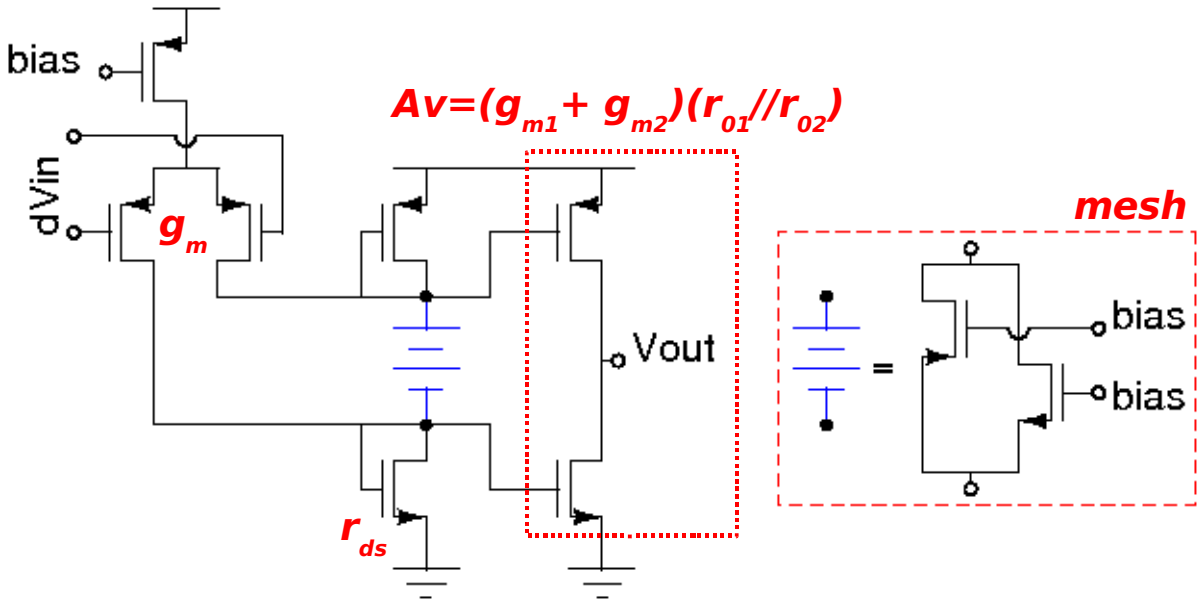
Shaper Core

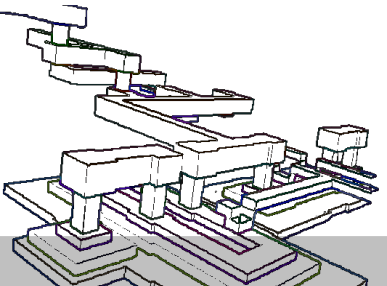
2nd Stage (1/2)





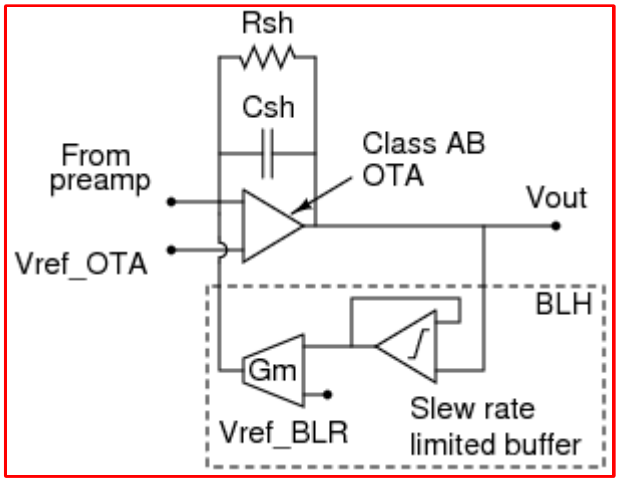
Peaking time	10-20 ns
Linear output swing	2.9 V
Nonlinearity	1.5 %
Slew rate @10 pF load	500 V/μs
Noise @10 pF capacitance	1450 e^-
Power consumption	3.3mW

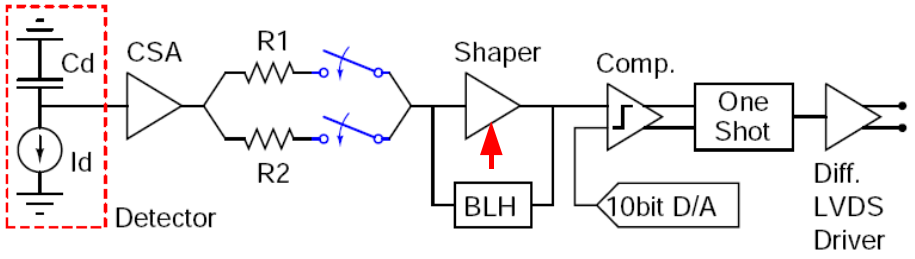




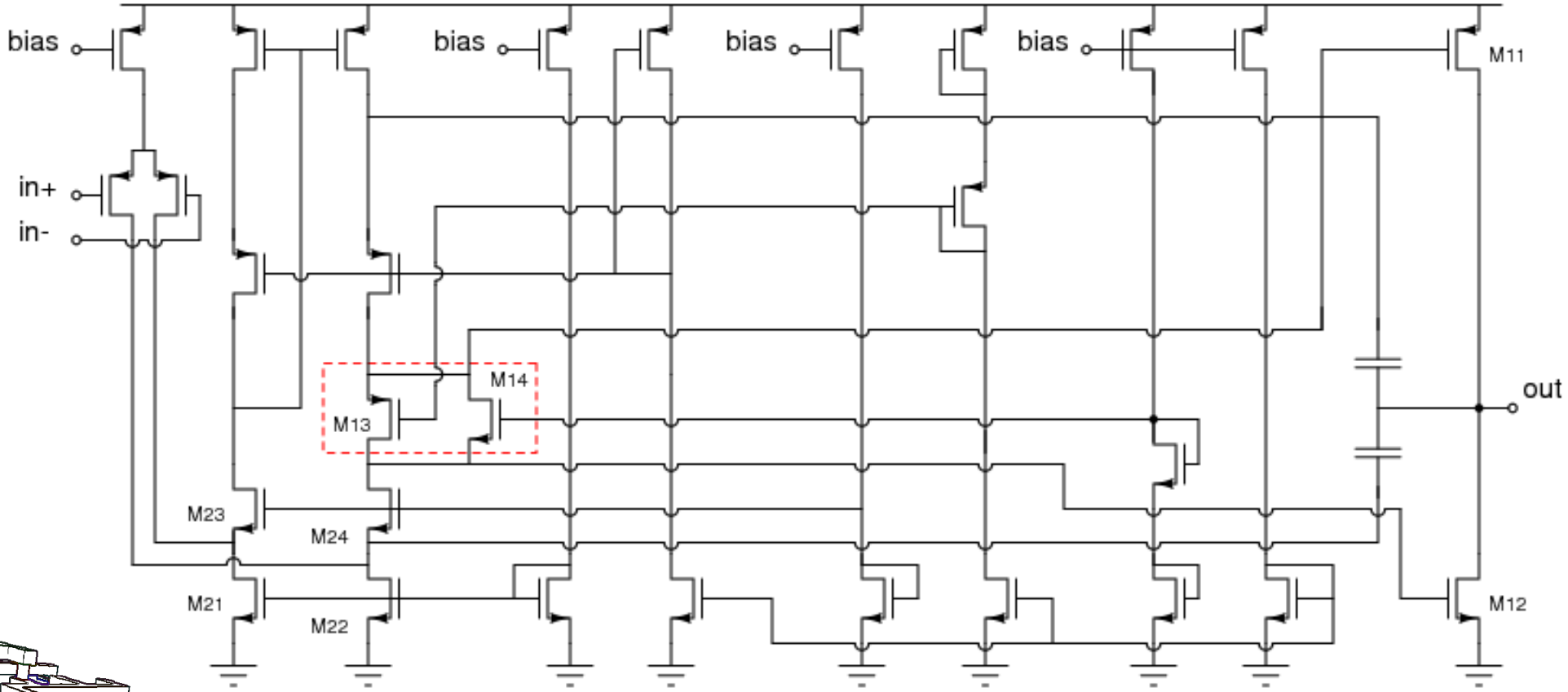
Blocks

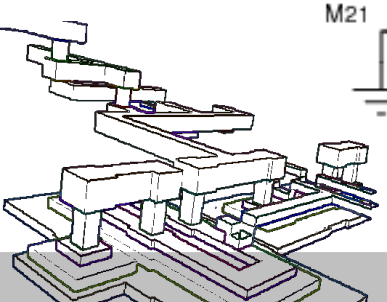
Shaper Core
2nd Stage (2/2)





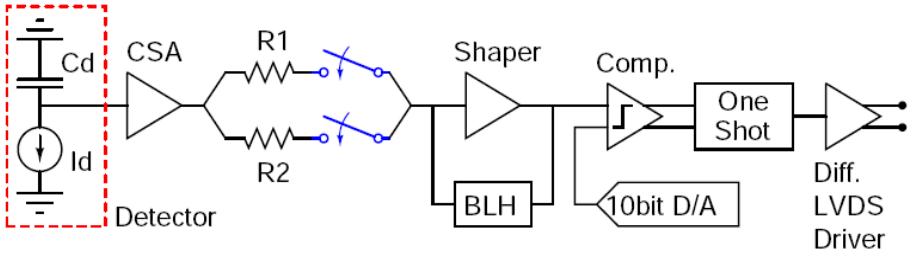
Peaking time	10-20 ns
Linear output swing	2.9 V
Nonlinearity	1.5 %
Slew rate @10 pF load	500 V/μs
Noise @10 pF capacitance	1450 e^-
Power consumption	3.3mW

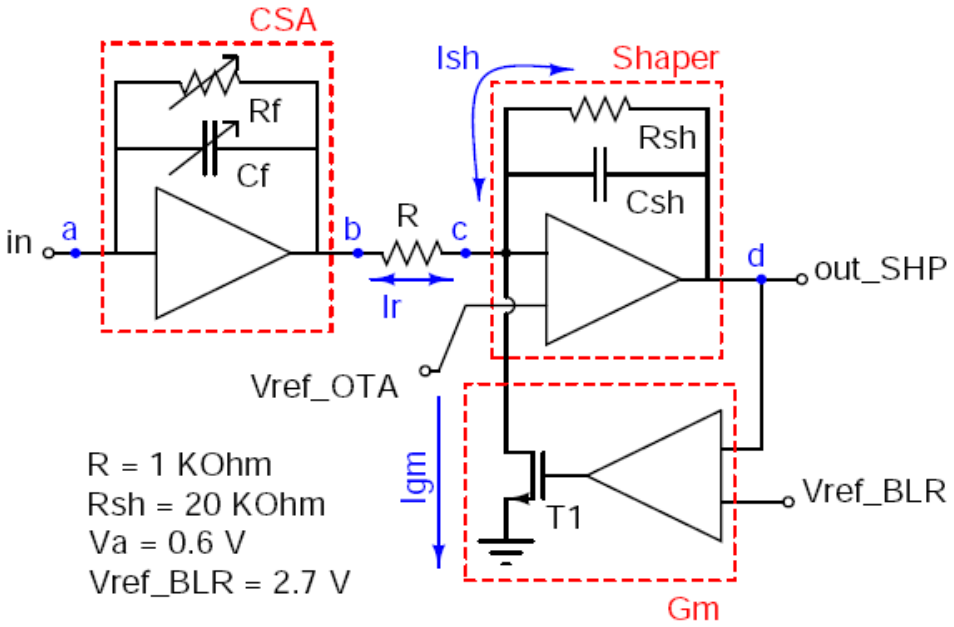


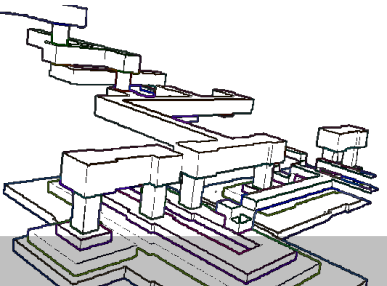


Blocks

FE Operation



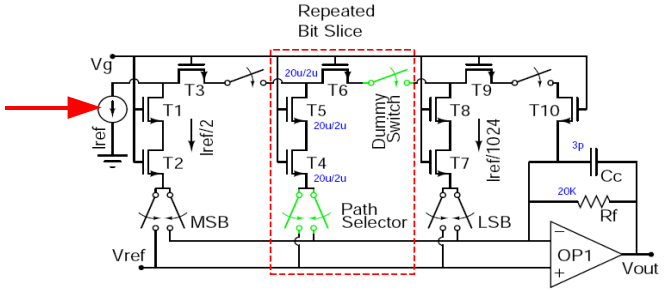


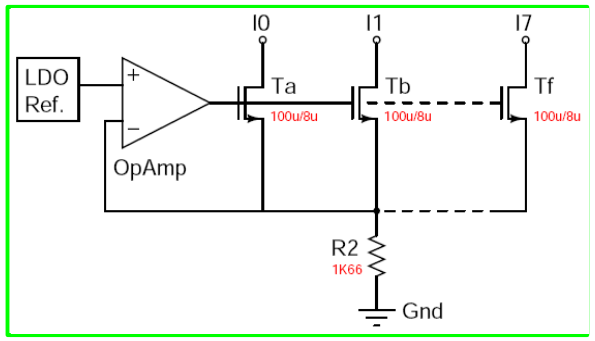


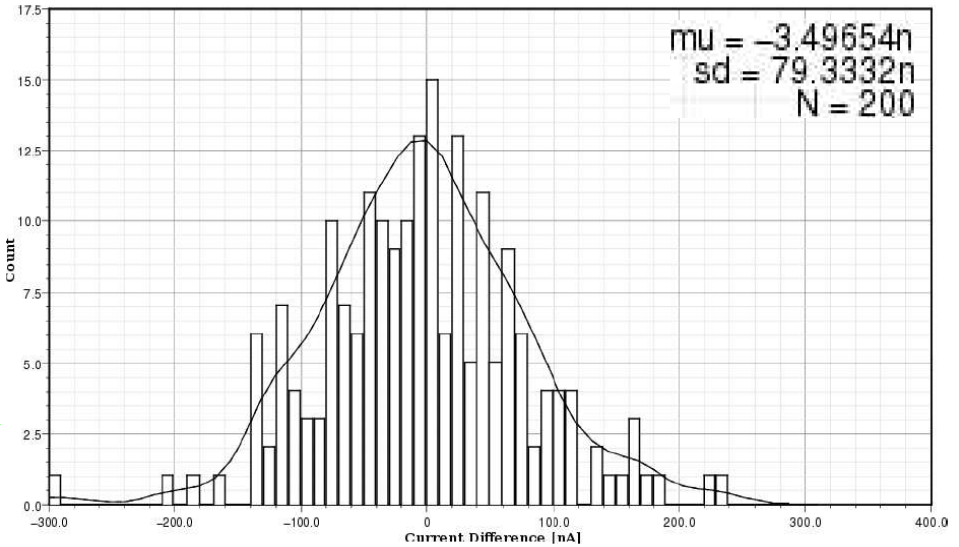
Blocks

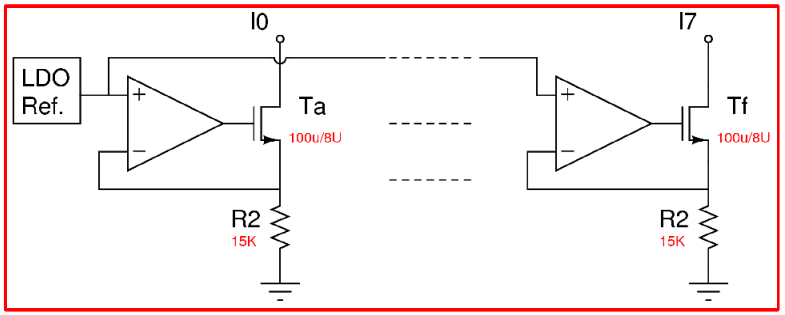
Current Sink (2/2)

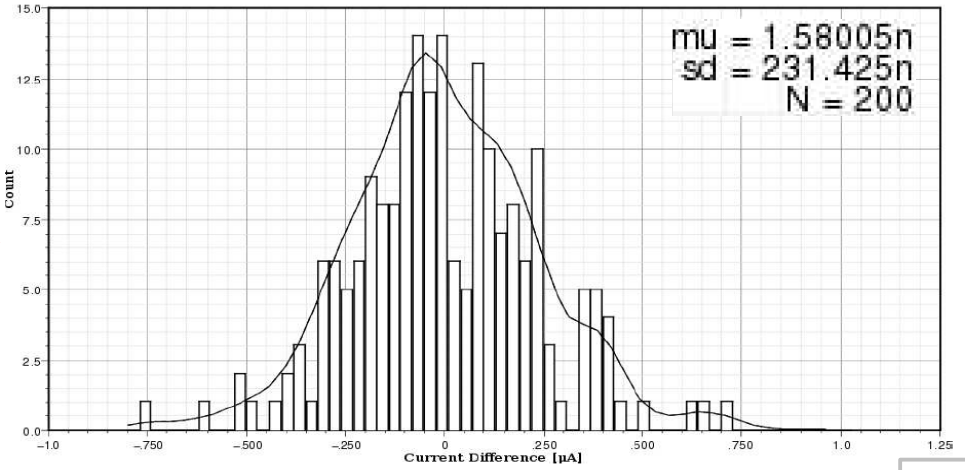
- Identical ***multi-channel*** architecture requiring identical biasing channels
- Rely on ***matching*** between :
  - Transistors *or*
  - Resistors
- MC showed : rely on transistor matching





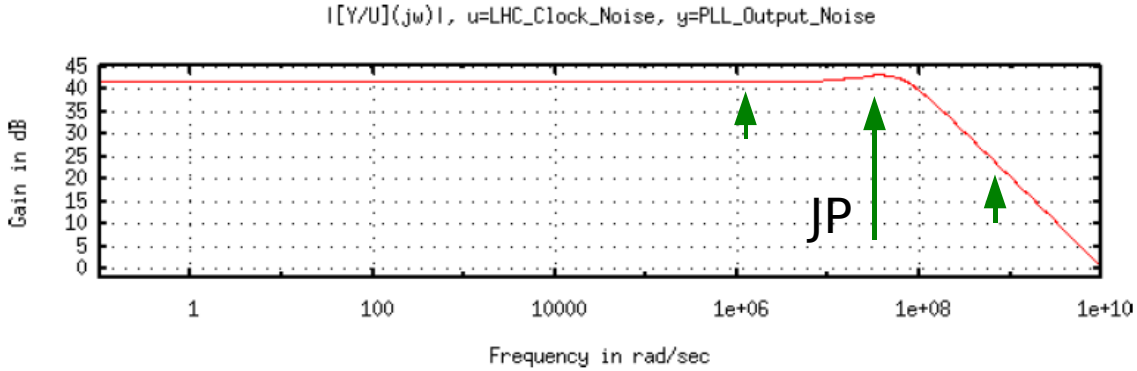


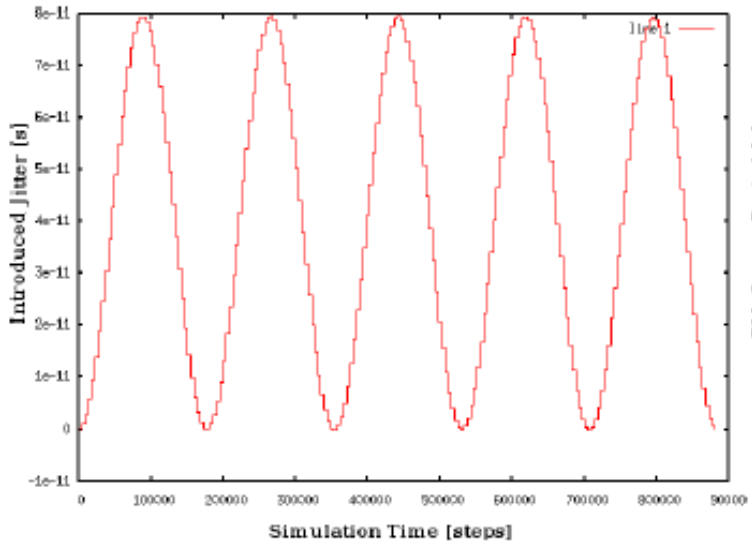


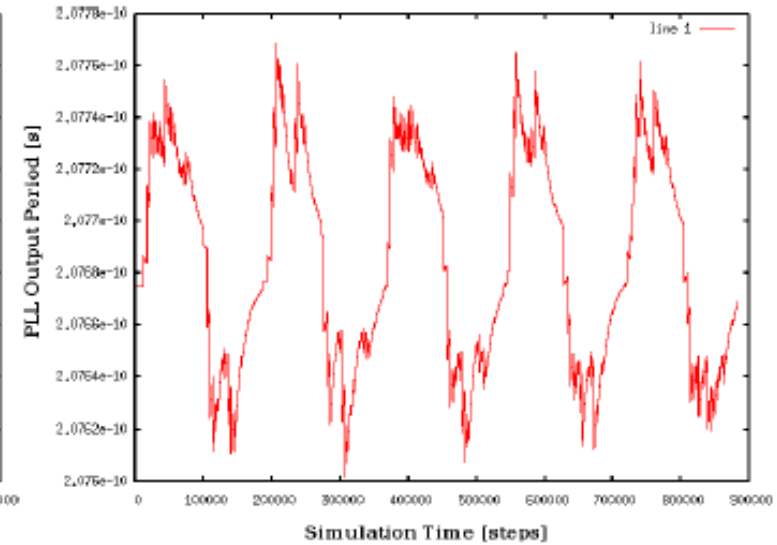


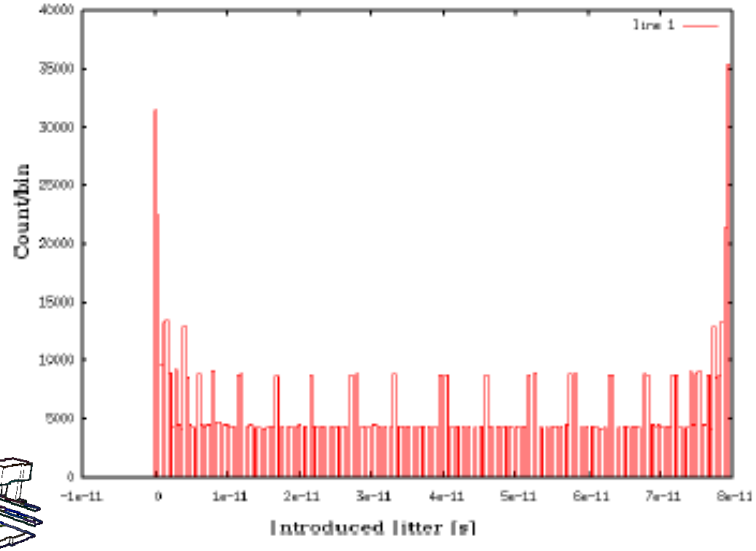
Verification (Octave m-files)

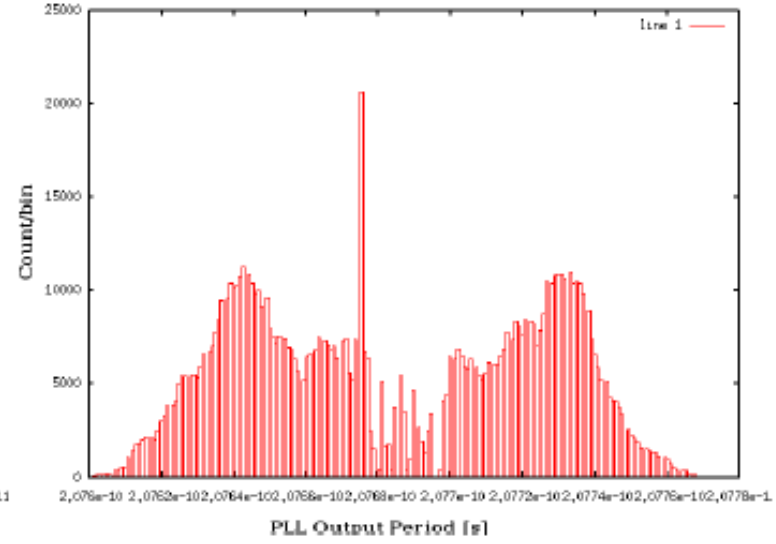
- Verify transfer functions of the ***loop***, the ***reference jitter*** to output and the ***VCO jitter*** to output
- ***Jitter peaking*** (JP) is important

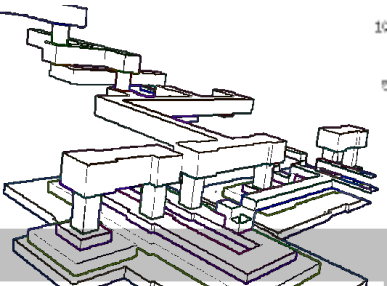




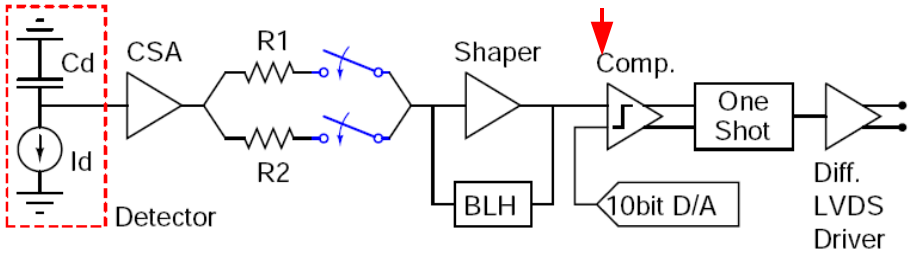


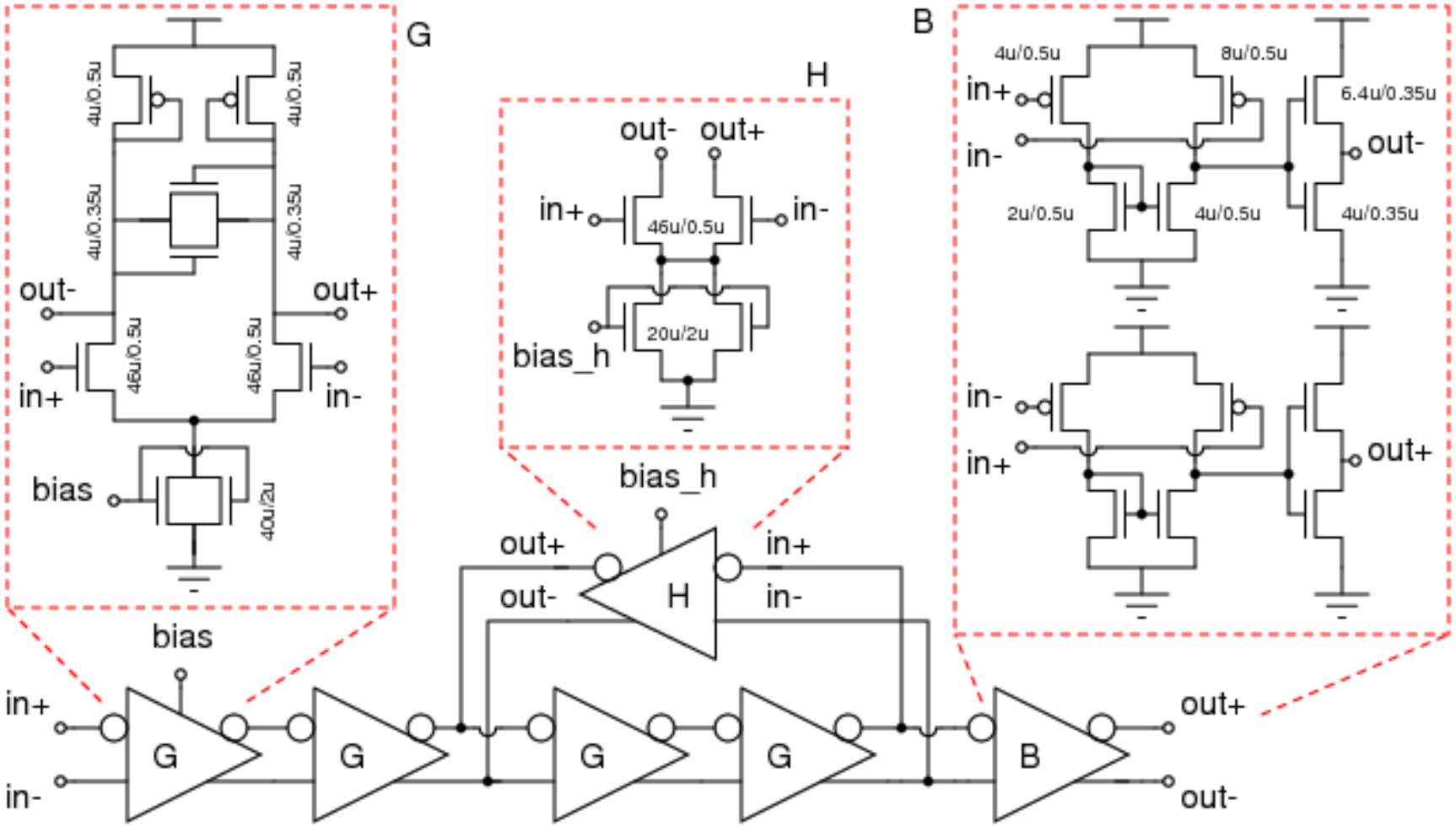


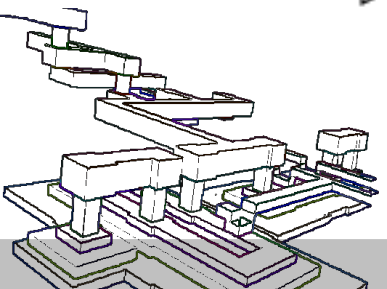




Blocks
Comparator







Implementation

Adopted CDR is a hybrid approach of:

- Gated-VCO, open-loop, burst-mode
- Filter-based, closed-loop, continuous-mode